

# **Robust Airline Fleet Assignment**

A Thesis  
Presented to  
The Academic Faculty

by

**Barry Craig Smith**

In Partial Fulfillment  
of the Requirements for the Degree of  
Doctor of Philosophy in Industrial and Systems Engineering

Georgia Institute of Technology  
August 2004

Copyright © 2004 by Barry Craig Smith

# **Robust Airline Fleet Assignment**

Approved by:

Dr. Ellis L. Johnson, Advisor

Dr. Earl R. Barnes

Dr. Anton J. Kleywegt

Dr. George L. Nemhauser

Dr. Timothy L. Jacobs  
(American Airlines)

August 12, 2004

## Acknowledgements

Many people have supported me through the years of school and research. While I appreciate them all, I can acknowledge only a few here. First, I'd like to thank the Georgia Tech ISYE department for giving me an opportunity to participate in their program as a non-traditional, geographically-challenged student in Texas. I've learned more from the faculty and fellow students than I ever expected. Special thanks to my friends in the Atlanta Admirals Club who made sure I never missed a flight. I'd like to thank Sabre for giving me the time to pursue a degree that too many people thought I already had. My dissertation committee has provided a great deal of guidance and encouragement; it has been a privilege to work with them.

Many people at Sabre provided invaluable support. In particular, I'd like thank Ladislav for giving me a clear perspective early in the program; Yanjun for help in algorithms; Stacy for help with Concert and C++; Tim, Michael and Dirk for guidance and many significant contributions to this research; finally Ross and Nancy for keeping the Sabre Research Group running smoothly during the times I was distracted with school and research.

I've had the great privilege of working with Ellis for eight years. It was his approach to problem solving that inspired me to go back to school. Although I have absorbed a very small part of what he has communicated, it has fundamentally changed my appreciation for the application of mathematics to a wide range of problems.

I'd like to thank my parents for creating a home environment that encouraged education and learning. I hope I can live up to the example that they set.

Finally, I'd like to thank my long-neglected wife Dianne for her unwavering support. I would not have made it through the highs and lows of this program without her encouragement and confidence, and I love her for it.

# TABLE OF CONTENTS

List of Tables	vi
List of Figures	viii
Summary	x
Chapter 1 Introduction	1
Chapter 2 Airline Fleet Assignment Modeling	4
2.1 Introduction	4
2.2 Development of Basic FAM	4
2.3 Basic FAM Model Formulation	8
2.3.1 Sets	8
2.3.2 Decision Variables	8
2.3.3 Parameters	8
2.3.4 FAM Formulation	9
2.3.5 Data	9
2.4 FAM Extensions	13
2.5 Robust FAM	16
2.5.1 Demand Robustness: Dynamic Scheduling	16
2.5.2 Operational Robustness	17
2.6 FAM Scenarios	18
Chapter 3 Adding Purity to FAM using Station Decomposition	20
3.1 Introduction	20
3.2 Station Purity	20
3.3 Station Decomposition	24
3.4 SDM Formulation	27
3.4.1 SDM Sets	27
3.4.2 Decision Variables	27
3.4.3 Parameters and Data	27
3.4.4 SDM Master Problem Formulation	28
3.4.5 Dual Variables	29
3.4.6 SDM Subproblem Formulation	29
3.4.7 Generating Plans from SDM <sub>sp</sub> Solutions	31
3.5 SDM Model Flow	32
3.5.1 Initial Plan Generation	32
3.5.2 SDM <sub>mp</sub> Solution	32
3.5.3 SDM <sub>sp</sub> Solution and Column Generation	33
3.5.4 Stopping Criteria	33
3.6 SDM Hub Selection	36
3.7 SDM Computational Results	41
3.8 Dual Improvement Approach to SDM	44

3.8.1 Solution Process	48
3.8.2 Computation Results	54
3.9 Integer Solutions	60
3.10 Improving MIP Performance with a Fix and Price Heuristic	63
3.11 Summary	66
Chapter 4 Revenue Management	67
4.1 Introduction	67
4.2 Notation	68
4.2.1 Sets	68
4.2.2 Decision Variables	68
4.2.3 Data and Parameters	68
4.3 Review of RM Literature	69
4.3.1 Overbooking	69
4.3.2 Seat Allocation	69
4.3.3 Impact of Seat Allocation on Flight Revenue	74
4.3.4 Flow Controls: Network Revenue Management	77
Chapter 5 Incorporating Revenue Management into FAM	81
5.1 Introduction	81
5.2 Literature Review	83
5.3 The ODFAMr Process	86
5.3.1 Improving ODFAMr LP Solutions	92
5.3.2 Getting MIP Solutions	97
5.4 ODFAMplr: ODFAM using Prorated Leg Revenue	100
5.4.1 ODFAMplr Formualtion	106
5.4.2 ODFAMplr Process Flow	107
5.4.3 ODFAMplr Results	107
5.4.4 ODFAMplr Timings	109
5.4.5 Summary of ODFAMplr Method	109
5.5 Summary of Revenue Management and FAM	109
Chapter 6 Family Purity, ODFAM and Station Decomposition	111
6.1 Introduction	111
6.2 Family Purity	111
6.3 ODFAM with Family Purity	116
6.4 Solving ODFAMplr with Purity using Station Decomposition	117
6.5 Penalty Costs	119
6.6 Summary	121
Chapter 7 Future Research	124
7.1 Improving ODFAM by Tightening Benders Cuts	124
7.2 Dantzig-Wolfe Approach to FAM	127
7.3 Airline Planning Integration	132
References	135

## LIST OF TABLES

Table 2.1	FAM scenarios, results and runtimes	18
Table 3.1	Impact of purity on US7	22
Table 3.2	Impact of purity on FAM performance	23
Table 3.3	Hub selection for US daily schedule	39
Table 3.4	Hub selection for International weekly schedule	39
Table 3.5	Impact of hub selection on SDMmp problem size	40
Table 3.6	Hub configurations for each schedule	40
Table 3.7	SDM performance for various network configurations	43
Table 3.8	SDMmp vs SSMP problem sizes	49
Table 3.9	Summary of SSMP results	56
Table 3.10	SSMP improvement details for US19 moderate purity	58
Table 3.11	FAM MIP performance	62
Table 3.12	SDM MIP performance	62
Table 3.13	SDM performance using fix and price heuristic	65
Table 4.1	Impact of seat allocation on flight revenue	74
Table 4.2	The impact of seat allocations versus capacity	75
Table 4.3	The impact of seat allocations with elastic demand	76
Table 4.4	The impact of seat allocations versus capacity	76
Table 5.1	Leg FAM versus ODFAM daily profit, US7	92
Table 5.2	ODFAMp solution weights for US7	95
Table 5.3	ODFAMp profit performance versus leg FAM and ODFAM	99
Table 5.4	Impact of changing capacity on the bid prices of other flights	105
Table 5.5	ODFAMplr results versus FAM and ODFAM, US7	107
Table 5.6	ODFAMplr results versus FAM and ODFAM, Int7	108
Table 5.7	ODFAMplr results versus FAM and ODFAM, US19	108
Table 5.8	CPU times required for ODFAMplr versus FAM, LP relaxation	109
Table 5.9	CPU times required for ODFAMplr versus FAM MIP	109
Table 6.1	Fleet and families for Star7 and Int7	113
Table 6.2	Fleet and families for US7 and US19	113

Table 6.3	Fleet and family impact on leg FAM	115
Table 6.4	Family purity with ODFAM	116
Table 6.5	ODFAM results with purity, single model vs. SDM	119
Table 6.6	The impact of family/station combination penalties	121
Table 6.7	Total annual benefits of ODFAM and purity	123

## LIST OF FIGURES

Figure 2.1	Sample timeline network	6
Figure 2.2	Aggregated timeline network	7
Figure 2.3	Expected revenue as a function of capacity and mean demand	12
Figure 3.1	Impact of purity on US7	23
Figure 3.2	Impact of purity on FAM CPU time	24
Figure 3.3	Star network	25
Figure 3.4	Multiple hubs	26
Figure 3.5	Station Groups	27
Figure 3.6	SDM model flow	35
Figure 3.7	Initial adjacency graph, $G_0$ with node degree	37
Figure 3.8	Adjacency graph $G_1$ after deleting first hub	37
Figure 3.9	Initial adjacency graph, $G_0$ with hubs and groups identified	38
Figure 3.10	The dual improvement algorithm on the $n$ th major iteration	50
Figure 3.11	$L(\pi)$ after iteration number 2.0	51
Figure 3.12	$L(\pi)$ after iteration number 2.1	51
Figure 3.13	$L(\pi)$ after iteration number 2.2	52
Figure 3.14	SSMP with Option 1, using all previous duals	53
Figure 3.15	SSMP with Option 3, using previous weighted dual	54
Figure 3.16	SSMP objective function value versus iteration	59
Figure 3.17	Optimality gap versus iteration using SSMP (US19)	59
Figure 3.18	SDM versus FAM CPU comparison	65
Figure 4.1	Nested class structure	73
Figure 4.2	Algorithm to solve ODYMr	78
Figure 5.1	Approximating the ODFAM revenue function on a single flight	87
Figure 5.2	ODFAM model flow	89
Figure 5.3	ODFAM revenue gap versus major iteration, US7	90
Figure 5.4	ODFAM profit versus leg FAM, US7	90
Figure 5.5	Revenue gap for ODFAMp, US7	94



Figure 5.6	ODYMp profit versus FAM, US7	94
Figure 5.7	Revenue Gap for approximate ODYMp	96
Figure 5.8	ODFAMp profit for approximate ODYMp	97
Figure 5.9	ODFAMp model flow	98
Figure 5.10	Algorithm to calculate prorated leg revenues	103
Figure 5.11	The impact of prorated leg revenue cuts	104
Figure 5.12	ODFAMplr model flow	107
Figure 6.1	Station decomposition with ODFAMplr	118
Figure 7.1	Adjusting a revenue cut to be tight at an integer solution	125
Figure 7.2	ODFAM model, FAM and ODYM as a single model	128
Figure 7.3	Dantzig-Wolfe decomposition of ODFAM	129
Figure 7.4	A summary of airline planning stages	133

## SUMMARY

Fleet assignment models are used by many airlines to assign aircraft to flights in a schedule to maximize profit. Major airlines report that the use of fleet assignment models increases annual profits by more than \$100 million. The results of fleet assignment models affect subsequent planning, marketing and operational processes within the airline. Anticipating these processes and developing solutions favorable to them can further increase the benefits of fleet assignment models.

We propose to produce fleet assignment solutions that increase planning flexibility and reduce cost by imposing station purity, limiting the number of fleet types allowed to serve each airport in the schedule. We demonstrate that imposing station purity on the fleet assignment model can limit aircraft dispersion in the network and make solutions more robust relative to crew planning, maintenance planning and operations. Because station purity can significantly degrade computational efficiency, we develop a solution approach, Station Decomposition, which takes advantage of airline network structure. Station Decomposition uses a column generation approach to solving the fleet assignment problem; we further improve the performance of Station Decomposition by developing a primal-dual method that increases the solution quality and model efficiency. Station Decomposition solutions can be highly fractional; we develop a “fix and price” heuristic to efficiently find integer solutions to the fleet assignment problem.

Airline profitability can be increased if fleet assignment models anticipate the effects of marketing processes such as revenue management. We develop an approach, ODFAM, which incorporates airline revenue management effects into the fleet assignment model. We develop an approach to incorporate station purity and ODFAM using a combination of column and cut generation. This approach can increase airline profit up to \$27 million per year.

Finally we propose areas of future research to improve fleet assignment model performance and to expand its scope by integrating other airline planning processes.

# **CHAPTER 1**

## **INTRODUCTION**

Fleet assignment models (FAM) are used by many airlines to assign aircraft to flights in order to maximize operating profit. Major airlines report that the use of fleet assignment models increases annual profit by over \$100 million (Abarra 1989) (Subramanian et al. 1994). FAM is part of a multi-step process that creates an airline's operating plan, refines it and ends with the plan's execution. The quality the plan and the profitability of an airline can be further improved if FAM anticipates and produces solutions favorable to subsequent planning and operational processes. The processes affected by fleet decisions include:

- Crew scheduling. FAM solutions with fleets serving a smaller number of stations with greater frequency provide more flexibility for crew assignment and can reduce crew costs.
- Aircraft maintenance. Airlines must provide equipment, spare parts and qualified mechanics to operate various aircraft types at each station. Limiting the diversity of aircraft serving each station can reduce maintenance costs.
- Marketing. The revenue generated by a schedule depends on how the airline's marketing department prices and sells the seats in the network. The quality of FAM's revenue estimates and solution depend on how well the model anticipates these marketing actions.
- Operations. Airline schedules are rarely operated exactly as planned. Airline operations are disrupted by weather, mechanical and ATC problems. As a result, aircraft and crews are reassigned from the original plan. The cost of recovering from schedule disruptions is affected by the opportunities to swap aircraft and or crew. Commonality in fleet types serving each station increases the opportunities

for swaps and can increase airline dependability and reduce the cost of disruptions.

Initial FAM formulations did not anticipate the impact of fleeting solutions on the subsequent planning and operational processes. FAM formulations have been investigated to address crew scheduling (Clarke et al. 1996), maintenance (Barnhart et al. 1998) and marketing (Jacobs et al. 1999) separately. We address the crew, maintenance and operational issues simultaneously through station purity, limiting the number of fleets that can serve each station. We address the marketing issues by modeling the impact of an airline's revenue management process. We demonstrate that both of these approaches have a significant negative impact on the efficiency of FAM. We develop a computational approach that is efficient enough to incorporate these approaches into relatively large FAM problems using a combination of column generation and cut generation. The solutions produced with this approach are robust relative to crew planning, aircraft maintenance, marketing and operations.

This dissertation is divided into 7 chapters. Chapter 2 provides an overview of the fleet assignment model. Chapter 3 describes station purity and its impact on FAM profit and efficiency. We develop a station decomposition approach to station purity that takes advantage of the hub and spoke structure typical of major airlines. Station decomposition uses a column generation approach that in some cases suffers from slow convergence. In order to increase solution efficiency, we develop a primal-dual approach that results in a significant reduction in the number of major iterations required to produce good solutions to the LP relaxation of the FAM problem. In order to produce integer solutions, we develop a procedure to fix certain assignment variables and then generate additional columns. This “fix and price” approach provides performance improvements compared to a single FAM model.

In Chapter 4 we provide an overview of airline revenue management modeling. We demonstrate the impact that revenue management can have on the quality of FAM solutions. In Chapter 5, we review approaches to incorporating revenue management into

FAM, typically known as ODFAM. We also develop and test several approaches using Benders decomposition. The best approach to solving the LP relaxation produces highly fractional solutions with very large MIP gaps. We develop a modification to this approach that significantly reduces solution fractionality and MIP gap.

In Chapter 6, we develop a formulation that combines ODFAM with station purity and demonstrate that on large problems station decomposition can produce better solutions more efficiently than solving FAM with a single monolithic model.

In Chapter 7, we suggest three areas of future research. First, we propose an approach to tightening Benders cuts in ODFAM in order to reduce fractionality of the LP solutions. Development of a general approach to reducing fractionality of Benders solutions could increase the applicability of this type of decomposition. Second, we propose a Dantzig-Wolfe decomposition to ODFAM that could produce better integer solutions. Finally, we review the airline planning process and suggest opportunities for the integration of models and processes that have been separate due to computational and modeling complexity.

## **CHAPTER 2**

### **AIRLINE FLEET ASSIGNMENT MODELING**

#### ***2.1 Introduction***

The airline Fleet Assignment Model, FAM, has been one of the most significant applications of linear and integer programming in the transportation industry. FAM has been credited at several airlines with increasing profits by more than \$100 million per year. In this chapter we review the development of basic FAM, provide details of basic FAM models and data, and review extensions to FAM to incorporate other aspects of planning and operations. Finally we review extensions to FAM to make the solutions more robust relative to operational and demand uncertainties.

#### ***2.2 Development of Basic FAM***

The airline fleet assignment problem has been a topic of academic and industrial interest for nearly 50 years. Ferguson and Dantzig (1955) formulate a combined fleet assignment and aircraft routing model that maximizes operating profit for a fixed schedule with known deterministic demand. One year later they revisit this problem with stochastic demand (Ferguson and Dantzig (1956)). Simpson (1978) develops a fleet assignment model that assigns aircraft to flights in order to minimize operating cost subject to constraints that all demand must be carried, aircraft movements at each station must balance and total available flying time must not be exceeded. While these models are interesting applications of large-scale optimization there is no indication that they are used by any airlines.

Following the deregulation of the US airline industry, the successful major carriers grew their flight schedules significantly and developed hub-and-spoke networks. The

concentration of flights into and out of hubs allowed airlines to serve many more origin and destination markets with connecting service. Flight scheduling problems that had been tackled manually prior to this expansion became much bigger and more complex. Abara (1989) publishes the first significant FAM application based on work done at American Airlines. Abara's model could either maximize operating profit or reduce operating cost by maximizing the utilization of the most efficient fleets. He develops a time-space network to track individual aircraft in the air and on the ground in order to ensure that the assignment of aircraft was physically feasible and to control their turns or routings. Abara defines a turn as the successive assignment of one aircraft to two flights; the aircraft "turns" from the arrival of the first flight to the departure of the second. Minimum turn times are required to allow for passenger and baggage unloading, cleaning, refueling, inspection, and passenger and baggage loading. To make the FAM/routing problem practical to solve, he limits the number of possible aircraft turns for each flight. Revenue is based on a stochastic model of demand, but Abara assumes that each leg is independent. Abara first solves the FAM LP relaxation, fixes variables and then solves the MIP. He reports that the solutions to the LP relaxation are largely integer. As a measure of solution quality, Abara observes that the number of high demand legs covered by large equipment increased from 76% to 90% and that the net profit impact of FAM is 1.4 margin points or \$105 million per year.

Hane et al. (1995) develops a formulation similar to Abara's but make significant computational improvements. The Hane model includes many of the fundamental elements common to subsequent FAM formulations. Their model minimizes operating costs, which includes the cost of spilling a potential passenger due to lack of capacity, subject to:

- Cover – every flight leg must be assigned exactly one fleet type
- Balance – aircraft cannot appear or disappear in the network
- Plane count – for each fleet, the total number of aircraft on the ground or in the air at any point in time cannot exceed the total available.

The Hane formulation includes a ground arc network that tracks aircraft on the ground, but unlike Abara, does not indicate aircraft turns. There is a timeline network for each airport, fleet type combination. Nodes indicate arrival and departure events. Arcs correspond to the assignment of equipment to a flight arrival, departure, or for some period on the ground. There is a return arc to ensure that the number of aircraft on the ground at the end of the time horizon (typically a day or week) is the same as at the beginning of the next time period. Figure 2.1 illustrates a sample timeline network with 3 arrivals, 3 departures, 5 ground arcs, and one return arc.

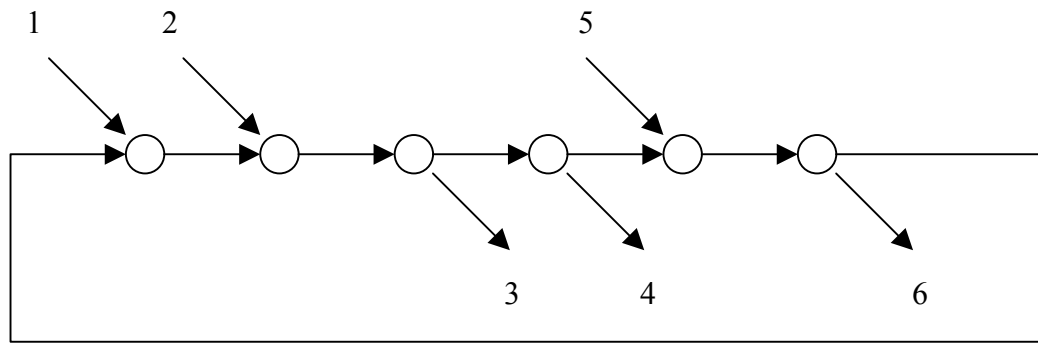


Figure 2.1: Sample timeline network.

There is a balance constraint associated with each node to ensure that aircraft do not appear or disappear from the network. Arc flows are non-negative to ensure that we don't assign more aircraft to departing flights than are present at this airport. There is a time specified in the network when we count aircraft either on the ground or in the air. This count is used to ensure that total plane counts are not exceeded.

The ground arc network accounts for a large portion of the overall fleet assignment problem size. The total number of balance constraints depends on the number of airports, fleets, and flights. Hane states that there are some arcs in the network that cannot have negative flow. For example in Figure 2.2, we can aggregate the nodes associated with flights 1,2,3 and 4 into one node. There is no feasible flow through this one aggregated node that is not feasible in the original network. As a result, we can reduce the number of ground arcs and balance constraints. Similarly, we can combine nodes for flight 5 and 6.



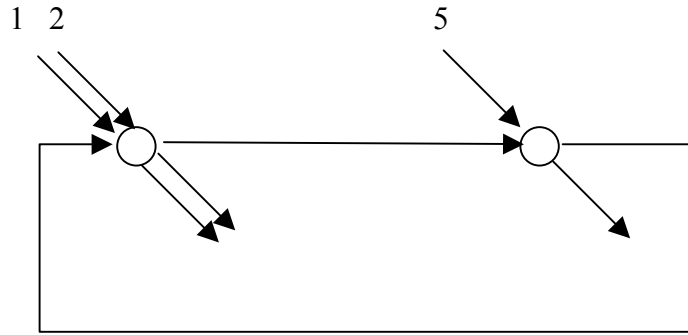


Figure 2.2: Aggregated timeline network.

Hane also identified situations in which specific ground arcs are very likely to have no flow. These ground arcs connect “islands” of flights. Unless there is significant spare aircraft capacity in the network, the islands can be considered isolated from the rest of the ground arc network. The isolation of islands reduces the number of ground arcs, but more importantly, it reduces the number of potential solutions, making the MIP easier to solve. While islands can significantly reduce problem complexity, their use can lead to sub-optimal solutions. In the example shown in Figure 2.2, it is possible to assign all flights in this network and zero flow on the return arc. In this case, the flow between the nodes is zero, and the two nodes could be considered separate islands. However, there may be situations where it is more profitable to have the aircraft assigned to flight 5 spend the night at this airport and then turn to flight 3 or 4 the next morning. If we assume that the two nodes form islands, we will miss this potentially beneficial solution.

Hane demonstrates that the combination of node aggregation, flight islands, the use of dual-steepest edge and MIP branching strategy can reduce run times by 2 orders of magnitude compared to the original FAM formulation. Subramanian et al. (1994) implemented a model with many of the features described by Hane at Delta Air Lines. They report an expected profit improvement of \$100 million per year.

## 2.3 Basic FAM Model Formulation

In this section, we describe the basic FAM model developed by Hane. Since the Hane paper was published, FAM notation has evolved. We use notation similar to that of Lahatepanont (2002).

### 2.3.1 Sets

$A$ : set of airports indexed by  $a$ .

$L$ : set of flight legs, indexed by  $i$ .

$F$ : set of fleet types, indexed by  $f$ .

$T$ : set of all departure and arrival events, indexed by  $t$ .

$CL(f)$ : set of flight legs crossing the counting line flow by fleet  $f$ .

$I(f, a, t)$ : set of flight legs inbound to  $\{f, a, t\}$ .

$O(f, a, t)$ : set of flight legs outbound from  $\{f, a, t\}$ .

### 2.3.2 Decision Variables

$$x_{f,i} = \begin{cases} 1 & \text{if fleet } f \in F \text{ is assigned to flight leg } i \in L \\ 0 & \text{otherwise} \end{cases}$$

$y_{f,a,t^+}$ : the number of aircraft on the ground for fleet type  $f \in F$ , at airport  $a \in A$ , on the ground arc just following time  $t \in T$ .

$y_{f,a,t^-}$ : the number of aircraft on the ground for fleet type  $f \in F$ , at airport  $a \in A$ , on the ground arc just prior to time  $t \in T$ .

### 2.3.3 Parameters

$N_f$ : the number of aircraft available of fleet type  $f \in F$ .

$Cap_f$ : the seating capacity of fleet type  $f \in F$ .

$C_{f,i}$ : operating cost of assigning fleet  $f \in F$  to flight leg  $i \in L$ .

$R_{f,i}$ : revenue produced by assigning fleet  $f \in F$  to flight leg  $i \in L$ .

### 2.3.4 FAM Formulation

(FAM)

Maximize:

$$\sum_{f \in F} \sum_{i \in L} (R_{f,i} - C_{f,i}) x_{f,i} \quad (2.1)$$

Subject to:

$$\sum_{f \in F} x_{f,i} = 1, \forall i \in L \quad (2.2)$$

$$y_{f,a,t^-} + \sum_{i \in I(f,a,t)} x_{f,i} - y_{f,a,t^+} - \sum_{i \in O(f,a,t)} x_{f,i} = 0, \forall f, a, t \quad (2.3)$$

$$\sum_{a \in A} y_{f,a,0^-} + \sum_{i \in CL(f)} x_{f,i} \leq N_f, \forall f \in F \quad (2.4)$$

$$x_{f,i} \in \{0,1\}, \forall f \in F, \forall i \in L \quad (2.5)$$

$$y_{f,a,t} \geq 0, \forall f, a, t \quad (2.6)$$

FAM maximizes operating profit: revenue minus operating costs, (Eq 2.1). Constraints 2.2, Cover, ensure that every flight leg is assigned exactly one fleet type. Constraints 2.3, Balance, ensure that the flow of aircraft in the timeline network is feasible. Constraints 2.4, Plane Count, ensure that the solution does not use more aircraft than are available. The decision variables associated with the assignments are binary; the decision variables associated with ground flow are non-negative. In any feasible solution, the ground flow is also integer.

### 2.3.5 Data

#### 2.3.5.1 Costs

In order to produce profit-improving solutions, FAM requires estimates of operating costs. These estimates include cost of flight crew, cabin crew, fuel, ground handling and maintenance. These costs are generally known within the airline and do not present any modeling challenges.

### 2.3.5.2 Revenue

Unlike costs, revenues are a challenge to estimate. Revenue for any flight is a function of demand, revenue per passenger, and capacity. Demand is stochastic. Estimating demand for any flight is subject to error due to random variations and forecasting errors.

Estimating actual demand is difficult since it is only partially observable. Airlines collect statistics on bookings. On flights for which demand exceeds capacity, bookings are truncated by capacity. As a result, historical booking patterns provide a biased view of true demand. Swan (1983) develops an approach to estimate true demand from historical booking patterns. Booking data from American Airlines and Swissair indicate that demand follows either a normal or gamma distribution. (The gamma distribution has better theoretical properties, no negative values; the normal distribution is easier to work with computationally). Swan develops models to estimate true demand based on observed booking patterns. The difference between true demand and booking is *spill*: customers who could not be accommodated on flight of first choice. Some of the spilled demand may be *reattributed* to other flights on the same airline. They are *recaptured* if space is available and they book an alternate flight.

The spill process is based on conditional expectation of the demand distribution truncated at capacity:

$$E(traf_i | dmd_i, cap_f) = P(dmd_i \leq cap_f)E(dmd_i | dmd_i \leq cap_f) + (1 - P(dmd_i \leq cap_f))cap_f \quad (2.7)$$

where,

$dmd_i$ : the demand distribution for flight leg  $i \in L$

$cap_f$ : the seating capacity for fleet  $f \in F$

$E(traf_i | dmd_i, cap_f)$ : the expected traffic for flight leg  $i \in L$  given the demand distribution for flight leg  $i$ ,  $dmd_i$  and seating capacity  $cap_f$ .

Swan develops a method to estimate the mean and standard deviation of the demand distribution from the mean of the traffic distribution and historical capacity.

The revenue coefficients used in FAM are typically expected revenue given a specified capacity. Given the demand distribution for flight leg  $i \in L$ , the expected revenue is found as:

$$R_{f,i} = E(traf_i | dmd_i, cap_f) RPP_i \quad (2.8)$$

where,  $RPP_i$ : is the average revenue per passenger for flight leg  $i \in L$ .

If we model demand with a gamma distribution, then estimating expected traffic is

relatively straight-forward:  $dmd_i \sim \Gamma(\alpha_i, \beta_i)$ , where  $\alpha_i = \left( \frac{Dmd\_std_i}{Dmd\_mean_i} \right)$  and

$\beta_i = \left( \frac{Dmd\_mean_i}{\alpha_i} \right)$ . From the definition of the gamma function (Drake 1967):

$$\Gamma(cap_i, \alpha_i, \beta_i) = p(dmd_i \leq cap_i) = \int_0^{cap_i} \frac{D^{\alpha-1} e^{-D/\beta}}{(\alpha-1)! \beta^\alpha} dD \quad (2.9)$$

The first term of Equation 2.7 can be evaluated as follows:

$$\begin{aligned} P(dmd_i \leq cap_f) E(dmd_i | dmd_i \leq cap_f) &= \int_0^{cap_i} D \frac{D^{\alpha-1} e^{-D/\beta}}{(\alpha-1)! \beta^\alpha} dD \\ &= \alpha \beta \int_0^{cap_i} \frac{D^\alpha e^{-D/\beta}}{(\alpha)! \beta^{\alpha+1}} dD \\ &= Dmd\_mean_i \Gamma(cap_i, \alpha_i + 1, \beta_i) \end{aligned} \quad (2.10)$$

In terms of the gamma distribution, Equation 2.7 becomes:

$$\begin{aligned} E(traf_i | dmd_i, cap_f) &= Dmd\_mean_i \Gamma(cap_i, \alpha_i + 1, \beta_i) \\ &\quad + (1 - \Gamma(cap_i, \alpha_i, \beta_i)) cap_f \end{aligned} \quad (2.11)$$

Expected traffic and revenue are concave functions of capacity. Figure 2.3 illustrates a case with three levels of mean demand and average revenue per passenger of \$100.

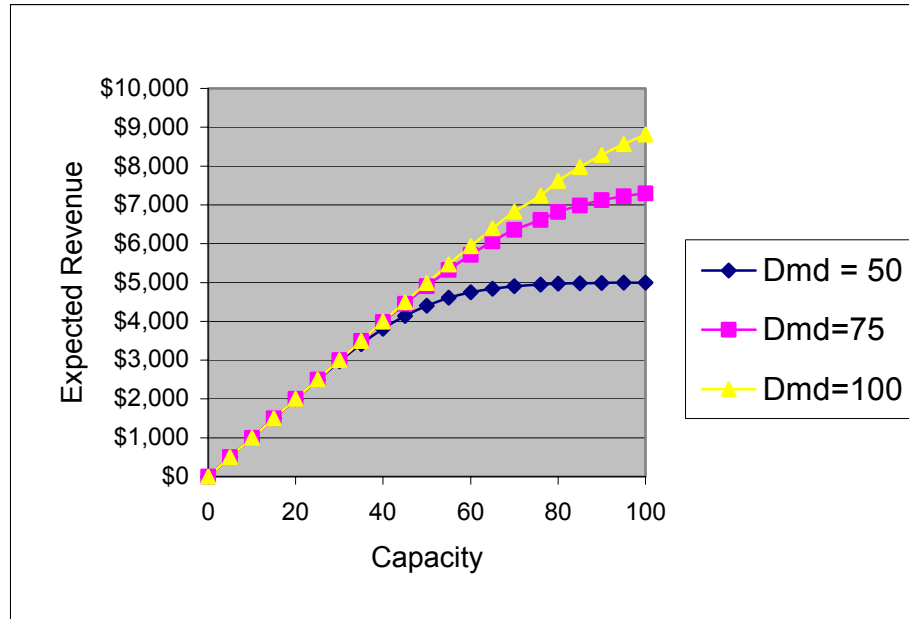


Figure 2.3: Expected revenue as a function of flight capacity and mean demand.

This approach to estimating revenue coefficients for FAM assumes that demand for flight legs is independent, and that all passengers for a flight have equal revenue. Since they have equal value, we also assume that all passengers are given equal access to reservations. Two major characteristics of the current airline environment make these assumptions less valid. First, due to the amount of connecting traffic in hub-and-spoke networks, the capacity decisions on one flight affects the revenues of others. Second, due to the wide range of fares offered in any given market, there is a great variation in the value of different passengers on any given flight leg. The revenue management (RM) process attempts to maximize total revenues given demand and capacity. The RM process can have significant impact on the revenues generated on any flight and in the entire network. Accurately modeling the effects of RM has been an active area of research since the mid 1990s. We review RM in more detail in Chapter 4 and its application to FAM in Chapter 5.

## **2.4 FAM Extensions**

While FAM added significant profit to AA and DL, the model scope is limited to the capacity planning process. FAM solutions have impact in many other airline planning and operational processes. In particular, FAM solutions may not meet maintenance requirements, and they may increase crew costs. Clarke et al. (1996) extends basic FAM to address both maintenance and crew. They add arcs to the FAM timeline network associated with maintenance events at required stations. Their formulation tries to avoid crew “lonely overnights.” Lonely overnights occur when a crew arrives at a station (not the crew’s base) late in the evening and its aircraft leaves before the crew has sufficient rest time. Clarke adds legal rest arcs to the network to ensure that the crew can stay with the aircraft. Both maintenance and crew modifications increase the size and complexity of the FAM problem.

Rushmeier and Kontogiorgis (1997) make the FAM formulation more flexible by replacing the time-space network with an event activity network that links successive activities that can be feasibly assigned to a single aircraft. This allows FAM to deal more directly with aircraft routing issues. Rushmeier and Kontogiorgis generalize the concept of flight islands to include connecting complexes such that turns for any flight are restricted to that complex. While turn-times in previous models was considered fixed based on the type of inbound and outbound flights, schedulers may cut corners with turn-times in order to reduce the number of aircraft required to operate a schedule. Rushmeier and Kontogiorgis relax the turn-time constraint and introduce a non-linear penalty to allow FAM to reduce turn-time when it can significantly reduce aircraft costs. They also introduce soft constraints with penalties to make the solution more maintenance and crew friendly. Rushmeier and Kontogiorgis report benefits to US Air of \$15 million per year.

Barnhart et al. (1998) incorporate aircraft routing directly into FAM. Where the Rushmeier and Kontogiorgis formulation deals with ensuring feasible aircraft turns in an activity network, Barnhart develops strings that represent the assignment of a sequence of flights to a single aircraft. A string is a sequence of connected flights that begins and ends at a maintenance base. Strings can incorporate all the complex rules associated with

FAA and carrier-specified maintenance. Because there are so many possible strings, in even a small schedule, Barnhart uses delayed column generation to produce the strings. The master problem is similar to basic FAM in that it minimizes the cost of selected strings subject to flight cover, flow balance and plane count constraints. Strings are generated using a shortest path approach in a *connection network* with costs based on dual values from the master problem. The nodes in the connection network represent flights; the arcs represent turns between the flights. A path in this network represents the assignment of one aircraft to flights (nodes in the path); the length of each path in the network corresponds to the reduced cost of this assignment. Using this approach, Barnhart is able to find maintenance feasible routing solutions to problems in both long-haul and short-haul networks.

Rexing et al. (2000) develop an approach to modifying scheduled departure and arrival times in the FAM solution process in order to reduce the number and cost of aircraft required to fly the schedule. They formulate FAM with a time window around each departure and add multiple copies of each flight spaced at 5-minute intervals through the window. The FAM cover constraint is applied to the flight copies so that only one of the flight copies is assigned capacity. This problem can be significantly larger than basic FAM. They use node aggregation and islands to reduce problem size; they also eliminate arcs associated with flight copies that become redundant after node aggregation. Since the reduced problem is still very large, they develop an iterative approach that adds flight copies only where the copies are beneficial to the solution. This approach begins by allowing any legal turn within the time windows. This solution may not be feasible to the original problem; they solve a sub-problem that adds flight copies in order to achieve feasibility to the original problem. While this method typically adds to the runtime, it significantly reduces problem size and memory requirements. On tests using a US major carrier's schedule, this approach retimes 8% of the flights and generates an additional profit of \$65,000 per day.

Gotz et al. (2001) develop a simulated annealing approach to the fleet assignment problem. Their approach is based on solution improvements in a local neighborhood



search. They demonstrate flexibility to deal with soft constraints as well as scalability to large problem sizes, up to 42,000 flight legs. Their approach can reduce computation time by up to 75%; solution quality in terms of profitability is typically within 0.5% of the optimum. Ahuja and Orlin (2002) develop a neighborhood search approach to integrating fleet assignment and aircraft routing models. They begin with separate optimal fleet and through assignment solutions. They identify swap cycles that improve the combined solution. In tests on a United Airlines schedule, their process runs in 6 seconds and produces an annual profit improvement of \$25 million compared to United's current process.

An important FAM extension is integration with schedule construction. Erdmann et al. (1999) develop an approach to schedule generation for European charter operations. They use a branch and price method to generate profitable aircraft rotations. Lettovsky et al. (1999) propose a method to either improve an existing schedule or to generate a schedule from scratch for scheduled airlines. Their approach uses column generation in which the columns represent the scheduled service plan for each origin, destination market pair. The non-linearities associated with the dependency of demand on competitive schedules are modeled in the service plan generation.

## **2.5 Robust FAM**

The FAM formulation discussed so far assumes that the schedule will be flown as planned. A schedule planned 90 days prior to departure is rarely flown exactly as planned. Controllable events such as schedule adjustments occur, patterns of demand can change, and finally operational disruptions occur. A plan that is optimal with respect to expected conditions may not be optimal or even feasible in actual operation. There are significant opportunities to improve performance of airline planning by ensuring that the plan is good (if not optimal) across a range of potential operations (Barnhart and Cohen 2002). In this section we review some aspects of robustness relative to uncertainty in demand and operations.

### **2.5.1 Demand Robustness: Dynamic Scheduling**

The solution quality of FAM depends on the accuracy of cost and revenue estimates. While cost estimates are relatively stable and well-known, revenue estimates depend on demand forecasts. Demand forecasts made early in the planning process, for example more than 90 days prior to departure, can have a significant error. Etschmaier and Mathaisel (1984) suggest that airline scheduling could be improved by delaying some fleet assignment decisions to take advantage of improved demand and revenue forecasts. Berge and Hopperstad (1993) develop an approach known as *Demand Driven Dispatch* ( $D^3$ ) that identifies capacity reassignments as departure date approaches.  $D^3$  assumes that aircraft swaps are made after crew scheduling is complete. As a result, only swaps within crew compatible aircraft families are allowed. Their formulation is similar to that of Hane; it uses a time-space network to control flow of aircraft. Due to problem size and potential fractionality of the solution, they develop two heuristics to solve the problem. First, they develop a *Sequential Minimum Cost Flow Method* to develop a FAM solution. Second, they develop a *Delta Profit Method* to find profit-improving swap cycles in a feasible solution. Both approaches perform well compared to optimal approaches with significant computation time savings. This approach assumes that flight times are fixed and that the  $D^3$  solution does not disrupt aircraft routings relative to maintenance constraints. Their testing indicates a potential for 1% to 5% profit improvements. Subsequent studies conclude similar benefits are possible (Waldman 1993, Cots 1999).

Airlines are adopting  $D^3$  but at a relatively slow pace (Feldman 2002). Continental Airlines reports benefits from their aircraft swapping process, Pastor (1999).

Listes and Dekker (2002) extend FAM to take advantage of dynamic scheduling. They develop a model whose objective is to determine the fleet composition that maximizes profit in a system that allows capacity swapping close to departure. They accomplish this with a 2-stage stochastic linear programming model. They generate multiple demand scenarios, and in the second stage, they solve a deterministic FAM model, for each of a set of these scenarios. The profit achieved in the second stage is used in the first stage model. In the first stage, they determine the fleet composition that maximizes the expected profit of the FAM solutions across all scenarios. They use a scenario aggregation approach (Rockafellar and Wets 1991) to improve the efficiency of the first-stage optimization process. They report a 90% runtime reduction due to scenario aggregation. They report profit benefits up to 0.5 margin points. Listes and Dekker model aircraft swapping in the second stage by determining aircraft assignments from a single demand scenario, generating new/modified demands and allowing swaps. While this procedure models some aspects of  $D^3$ , the initial assignments are based on a single scenario and are not robust relative to variations in demand. As a result, the benefits of  $D^3$  versus FAM may be understated.

List et al. (2003) develop an approach to robust fleet planning for the trucking industry. They also use a 2-stage stochastic optimization process. They argue that maximizing expected profit does not necessarily provide the most robust solution. They identify an acceptable threshold of performance, such as largest acceptable cost or minimum acceptable profit, and attempt to maximize the probability of achieving or surpassing this goal. While they solve relatively small problems compared to FAM, they provide a framework for determining the trade-offs between expected outcomes and risk.

### **2.5.2 Operational Robustness**

Airline operations are frequently disrupted by unplanned events such as airport capacity reductions due to weather, ATC delays and mechanical problems. Depending on the

number and severity of the disruptions, significant percentage of airline flight operations can be affected. For example, in December 2000, 33% of all US flight arrivals were delayed by more than 15 minutes (Mead 2003). FAM solutions can affect the time and cost required to return to planned operations. Bian et al. (2003) show that KLM's performance, as measured by departure and arrival delay probabilities is highly dependant on the number of aircraft on the ground at KLM's hub (Schiphol). Ageeva (2000) develops an aircraft routing model that encourages overlapping routes in the solutions so that aircraft have more swap opportunities in case of operational disruption. Rosenberger (2001) develops a FAM formulation to increase operational robustness by reducing hub connectivity. Rosenberger demonstrates through simulation that FAM solutions with decreased hub connectivity results in fewer cancellations and delays. Kang (2003) develops a strategy to layer the schedule into relatively independent sets of flights so that operational disruptions can be dealt with in one layer without spreading to the others. Kang shows, through simulation, that robust solutions significantly reduce passenger delays and disruptions compared to traditional FAM solutions.

## **2.6 FAM Scenarios**

A set of 4 scenarios is used to test the impact of adding robustness to FAM. These 4 scenarios are based on 2 schedules. The first is a weekly schedule for a mid-sized international carrier; the second is a daily schedule for a US domestic carrier. We construct a pure star network from the international schedule by deleting all flights that do not operate to or from a single designated hub. As we discuss in later sections, station decomposition should work most effectively on a network with a single hub in which all flights are to or from the hub. The international and US domestic schedules have 7 and 19 fleet types, respectively. We construct a simplified US domestic scenario by aggregating fleets into 7 crew-compatible families.

Table 2.1 summarizes the scenarios in terms of number of cities, number of flights, number of fleet and the number of possible fleet to flight assignments. Table 2.1 also summarizes the FAM problem size in terms of number of rows and columns in the LP and the CPU time required to solve the FAM LP relaxation and the MIP. These results

indicate a significant variation in time required to solve FAM based on problem size. In particular, the number of fleets has a significant impact on runtime.

Table 2.1: FAM scenarios, results and runtimes.

Case	Cities	Flights	Fleet Types	Legal Assignments	Rows	Cols	Profit*	Time**	
								LP	IP
Star7	44	1568	7	4991	4747	8163	65.38	6.30	2.31
Int7	50	2358	7	6537	6900	11072	82.54	12.64	7.24
US7	210	4182	7	27698	16547	40056	17.52	11.83	0.52
US19	210	4182	19	71096	35899	102794	19.36	253.00	8.53

\* \$ x 1,000,000

\*\* CPU seconds

All the tests are conducted on a Pentium 4 processor (2.0 GHz, 1.5 g RAM) using ILOG CPLEX 8.0 (ILOG 2001). The models are formulated in ILOG Concert 1.2 (ILOG 2000).

## CHAPTER 3

# ADDING PURITY TO FAM USING STATION DECOMPOSITION

### ***3.1 Introduction***

In this chapter we investigate the impact of station purity on FAM solution quality and runtimes. First, we add purity to the basic FAM formulation and demonstrate its impact on FAM efficiency. We then develop a station decomposition approach that improves FAM efficiency in cases with station purity. We investigate a primal-dual method to further improve the performance of the station decomposition approach. Finally, we develop a heuristic to efficiently produce integer solutions using station decomposition.

### ***3.2 Station Purity***

Station purity ensures that the number of fleet types serving a given station does not exceed a specified limit. This limit on the number of fleets serving a station is the station's purity level. If fleets in FAM are defined as crew-compatible families, then station purity ensures that there are opportunities to swap aircraft or crews for either operational or profitability reasons. Limiting the number of different fleets or families serving a station creates more opportunities for move-up crew assignments to cover operational disruptions.

Implementing station purity requires that we count the number of fleet types serving each station and add constraints on the total number of fleets for each station affected.

We add an auxiliary variable  $w_{f,s}$  to indicate whether fleet  $f \in F$  serves station  $s \in A$  in the FAM solution, and limit the number of fleets for each station by adding the following new constraints to the basic FAM formulation.

$$w_{f,s} \geq x_{f,i} \forall f \in F, s \in A, i \in L \quad (3.1)$$

$$\sum_{f \in F} w_{f,s} \leq SP_s \forall s \in A \quad (3.2)$$

$$w_{f,s} \in \{0,1\} \forall f \in F, s \in A \quad (3.3)$$

We investigate two levels of station purity: Maximum purity with  $SP_s = 1$  for all spokes; Moderate purity with  $SP_s = 1$  for small spokes and  $SP_s = 2$  for large spokes. Some stations have flights that require specific aircraft types for operational reasons. Due to these flight assignment restrictions, it is not always possible to limit the number of fleets to 1 or 2. Let  $SPMin_s$  be the minimum number of fleets required to serve spoke  $s$ . In the Maximum Purity case  $SP_s$  is set to the minimum feasible number for each spoke station.

$$\text{Maximum Purity: } SP_s = \max(1, SPMin_s) \quad (3.4)$$

In the Moderate Purity case,  $SP_s$  for small spokes is unchanged, for larger spokes,  $SP_s$  is at least 2.

$$\text{Moderate Purity: } SP_s = \max(1, SPMin_s) \text{ for small spokes} \quad (3.5)$$

$$SP_s = \max(2, SPMin_s) \text{ for large spokes} \quad (3.6)$$

In the tests conducted in this study, small spokes are defined as having less than 20 operations in the planning horizon (daily or weekly). Large spokes typically have less than 100 and 60 operations for the daily and weekly scenarios, respectively. Stations with operations in excess of these cutoffs are given no purity constraints in any scenarios. Robustness is improved when every fleet serving a station operates at relatively high frequency into and out of that station. This has two benefits. First, by limiting the

number of aircraft types that can serve any non-hub station, purity reduces the total number of stations served by any aircraft type. This reduction is illustrated in Figure 3.1 for the US domestic schedule with 7 equipment types (US7). For example, the 737 fleet serves over 119 stations in the base case, 81 in the moderate purity case, and 63 in the maximum purity case. There is a significant reduction in the number of spoke stations served by every fleet type, except the 727. The 727 is the least profitable fleet in the schedule. Imposing purity constraints forces the 727 to be flown slightly more frequently, as a result there is a slight increase in the number of stations served. The second impact of purity is to reduce the number of lonely fleets and fleet/station combinations in the solution. Lonely fleets, are defined by situations in which a fleet serves a station once or twice during the planning horizon (daily or weekly). Table 3.1 summarizes the impact of purity for the US7 schedule.

Table 3.1: Impact of purity on US7.

Purity	Singletons	Total Combinations
Base	164	630
Moderate	33	357
Maximum	22	280



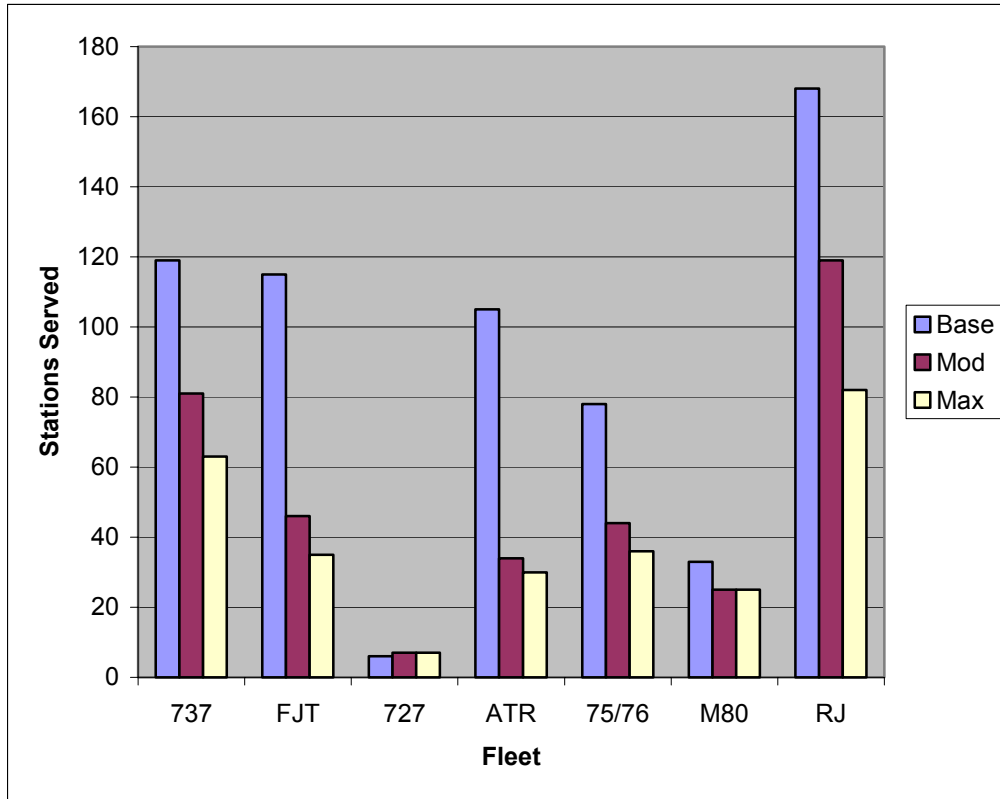


Figure 3.1: Impact of purity on US7.

Table 3.2 summarizes the impact of station purity on profit and runtime. Maximum purity reduces profit by up to 11%; runtimes increase by up to 500%. Moderate purity has a smaller impact on profit and runtimes. The impact of purity on runtimes is illustrated in Figure 3.2.

Table 3.2: Impact of station purity on FAM performance.

FAM	Base		Max Purity		Moderate Purity	
	Profit	Time	Profit	Time	Profit	Time
Star_7	65.38	6.3	64.18	23.8	65.31	13.5
Int_7	82.55	12.6	81.43	45.4	82.42	31.6
US_7	17.52	11.8	14.67	184.5	16.67	91.2
US_19	19.36	253	15.84	8306	18.17	2019

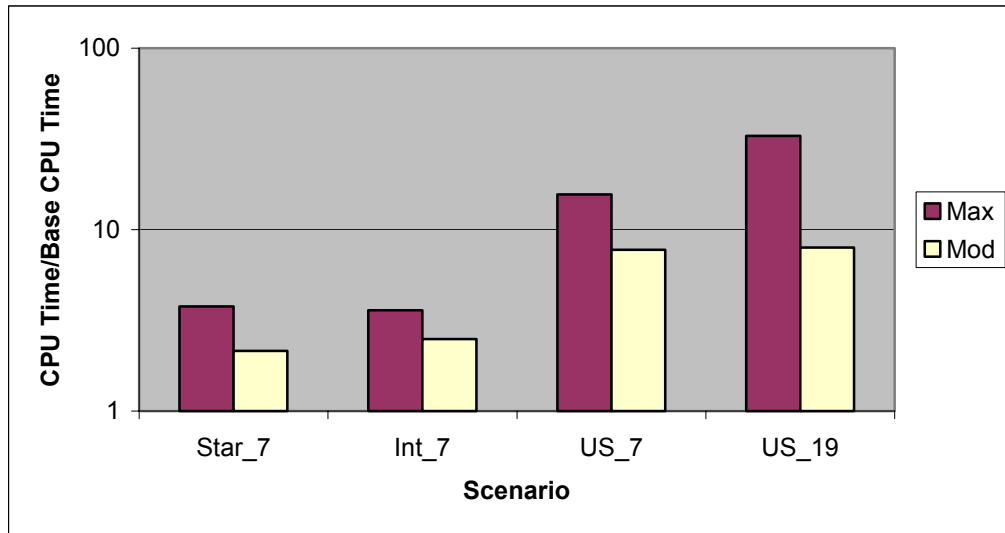


Figure 3.2: Impact of purity on FAM CPU time.

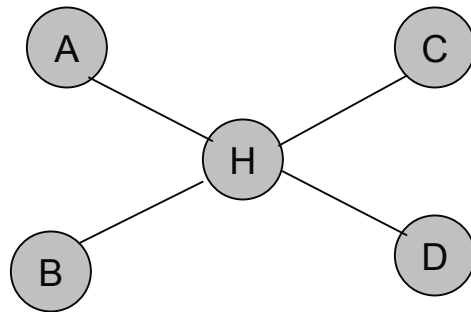
### 3.3 Station Decomposition

Station Decomposition takes advantage of the hub and spoke structure of typical airline flight networks. Airlines use hub and spoke networks in order to increase the number of passenger origin-destination markets (O&Ds) that can be served with a given number of flights. In some networks, almost all flights operate either to or from a small number of hubs. As a result, traffic and revenue for a single flight leg depends not only on the capacity for this leg but also on the capacities of upline and downline flights serving passengers in connecting markets.

If we remove the hubs in a pure hub and spoke network, the network decomposes into a set of spoke stations, each with its own set of flights. Because the number of flights operating to and from any spoke is a small part of the full schedule, determining fleeting solutions for each spoke is relatively easy. We refer to each feasible solution for a spoke as a fleeting plan. A solution to the full network fleet assignment problem is a collection of plans (one for each spoke) that satisfies aircraft count and flow balance at the hub. In

this approach, a single variable accounts for all decisions associated with fleet assignments for a spoke. This variable expansion approach has been shown to provide tighter LP relaxations, and is the basis for crew scheduling models. (Barnhart and Cohen 2002, Vanderbeck 2000).

Consider a star network containing one hub with flights to and from 4 spoke stations (Figure 3.3). In the Station Decomposition Model (SDM) structure, the master problem maximizes operating profit for all plans, subject to balance at H, plane count and cover constraints for the plans. There is a subproblem for each of the 4 spoke stations. For example, the subproblem for spoke A maximizes reduced cost for flights to and from A subject to balance at A, and cover for flights between H and A.



3.3: Star Network.

Airline schedules are more complex than a simple star network; they generally include multiple hubs and flights that operate between spokes. The SDM formulation assumes that all flights operate between a hub and spoke or between hubs; no spoke to spoke flights are allowed. We have flexibility in determining which cities should be designated as hubs in order to meet these requirements. As we will demonstrate, the choice of hubs determines the size of the master and subproblems and as a result has significant impact on the efficiency of this formulation.

There are two approaches to making a general network fit into the SDM structure. First, multiple stations can be identified as hubs. Second, spoke stations can be grouped

together and modeled as a single unit. Suppose there are flights between stations A and B in the example network. If we designate station A as a hub (Figure 3.4), then the master maximizes operating profit for all plans, subject to balance at H and A, plane count and cover constraints for the plans, and for flights operating between A and H. There is a subproblem for each of the 3 spoke stations. The subproblem for spoke B maximizes reduced cost for flights to and from B subject to balance at B, and cover for flights operating between H, A and B. Note that flights operating between A and H, hub-to-hub (H2H) flights, are not part of any plans. These H2H flights are modeled in the SDM Master identical to the FAM formulation in Equations 2.1-2.6. As more stations are designated as hubs, the number of H2H flights and the size of the SDM Master both increase.

The alternative to adding A to the master is to combine stations A and B into a station group (Figure 3.5). In this case, the original master problem is unchanged, the subproblems for A and B are combined. Again, the master maximizes operating profit for all plans, subject to balance at H, plane count and cover constraints for the plans. There are 3 subproblems: one for the group containing A and B, and one for each of the single station groups at C and D. The subproblem for group g maximizes the reduced cost for flights to, from, and operating within g subject to balance at all stations in g and cover for flights operating to, from, and within g. As more spokes are grouped together, the size of the subproblems increases.

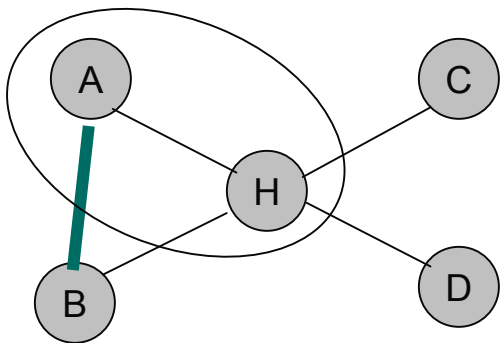


Figure 3.4: Multiple Hubs

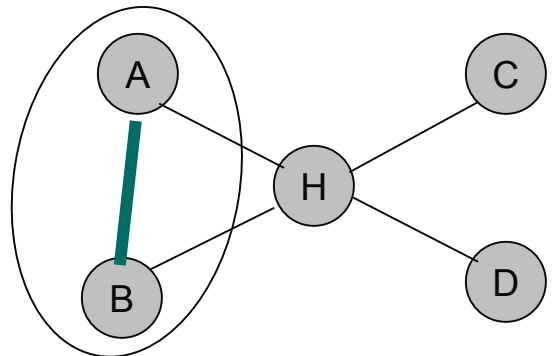


Figure 3.5: Station Groups

The process to determine which stations are hubs and which stations are grouped is discussed in Section 3.5. The master and subproblem formulations are discussed in detail in the next section.

### 3.4 SDM Formulation

SDM consists of a master problem and a subproblem for each station group. In the following sections, we define the additional sets and decision variables for SDM and the formulation for the master and subproblems.

#### 3.4.1 SDM Sets

$H$  : set of hub airports, indexed by  $h$ .  $H \subset A$ .

$S$  : set of spoke airports, indexed by  $s$ .  $S \subset A$ ,  $H \cap S = \emptyset$ .

$G$  : set of station groups, indexed by  $g$ .

$A^g$  : set of airports in station group  $g \in G$ , indexed by  $a$ .

$P$  : set of assignment plans, indexed by  $p$ .

$P^s$  : set of plans for spoke  $s$ , indexed by  $p$ .

$P^g$  : set of plans for station group  $g \in G$ , indexed by  $p$ .

$L^g$  : set of flight legs within group  $g \in G$ , indexed by  $i$ .  $L^g \subset L$

$L^h$  : set of hub-to-hub flight legs  $h \in H$ , indexed by  $i$ .  $L^h \subset L$ ,  $L^h \cap L^g = \emptyset$ .

#### 3.4.2 Decision Variables

$$x_p = \begin{cases} 1, & \text{if plan } p \text{ is in the SDM solution} \\ 0, & \text{otherwise} \end{cases}$$

$$x_{f,i}^p = \begin{cases} 1, & \text{if leg } i \in L \text{ is assigned fleet } f \in F \text{ in plan } p \in P \\ 0, & \text{otherwise} \end{cases}$$

#### 3.4.3 Parameters and Data

$$R_p : \text{revenue for plan } p, \quad R_p = \sum_{f \in F} \sum_{i \in L} R_{f,i} x_{f,i}^p.$$

$C_p$ : cost for plan  $p$ ,  $C_p = \sum_{f \in F} \sum_{i \in L} C_{f,i} x_{f,i}^p$ .

$PC_f^p$ : the number of aircraft from fleet  $f \in F$  on the ground or in the air at the counting time in plan  $p \in P$ .

$$q_{f,h,t,p} = \begin{cases} 1, & \text{if plan } p \text{ includes an arrival of aircraft type } f \text{ at hub } h, \text{ time } t \\ -1, & \text{if plan } p \text{ includes a departure of aircraft type } a \text{ at hub } h, \text{ time } t \\ 0, & \text{otherwise} \end{cases}$$

### 3.4.4 SDM Master Problem Formulation (SDMmp)

Maximize:

$$\sum_{f \in F} \sum_{i \in L} (R_{f,i} - C_{f,i}) x_{f,i} + \sum_{p \in P} (R_p - C_p) x_p \quad (3.7)$$

Subject to:

$$\sum_{f \in F} x_{f,i} = 1, \forall i \in L^h, \forall h \in H \quad (3.8)$$

$$y_{f,h,t^-} + \sum_{i \in I(f,h,t), i \in L^h} x_{f,i} - y_{f,h,t^+} - \sum_{i \in O(f,h,t), i \in L^h} x_{f,i} + \sum_{p \in P} q_{f,h,t,p} x_p = 0, \forall f, h, t \quad (3.9)$$

$$\sum_{h \in H} y_{f,h,0^-} + \sum_{i \in CL(f), i \in L^h} x_{f,i} + \sum_{p \in P} PC_f^p x_p \leq N_f, \forall f \in F \quad (3.10)$$

$$\sum_{p \in P^g} x_p = 1, \forall g \in G \quad (3.11)$$

$$x_{f,i} \in \{0,1\}, \forall f \in F, \forall i \in L^h \quad (3.12)$$

$$x_p \in \{0,1\}, \forall p \in P \quad (3.13)$$

$$y_{f,h,t} \geq 0, \forall f, h, t \quad (3.14)$$

The SDM master is similar to the basic FAM model with the replacement of some spoke related decision variables and constraints by the addition of plans. We include the revenue and cost associated with plans in the objective function (3.7). These revenues and costs are the accumulated values for individual flights in the plans. Equations 3.8 are

the cover constraints for flights that are not in the plans, the H2H flights. Equations 3.9 are the balance constraints at the hubs. The first four terms are identical to the base FAM formulation, although they apply only to H2H flights. The last term provides the incidence of flights to and from each hub for each plan. Equations 3.10 are the plane count constraints. The first two terms count the aircraft assigned to H2H flights crossing the counting line. The third term sums the number of aircraft crossing the counting line in each plan. Equations 3.11 are the convexity constraints on the plans; they ensure that each station group has one and only one plan assigned. Equations 3.12 ensure that the decision variables for H2H flights are binary. Equations 3.13 ensure that the decision variables associated with the plans are binary. Equations 3.14 ensure that ground flow at the hubs is non-negative.

### 3.4.5 Dual Variables

We use three sets of dual variables in the plan generation subproblem:

- $\pi_g^{conv}$  : dual variable on the convexity constraint for group  $g \in G$ .
- $\pi_f^{pc}$  : dual variable on the plane count constraint for fleet  $f \in F$ .
- $\pi_{f,h,t}^{bal}$  : dual variable on the balance constraint for fleet  $f \in F$ , hub  $h \in H$ , node  $t \in T$ .

### 3.4.6 SDM Subproblem Formulation (SDMsp)

The plan generation subproblem is solved for each station group,  $g \in G$ . Plans are generated in each iteration based on the value of the dual variables from the master (SDMmp). For each spoke station, a network of arrival, departure events and ground activity is created. These networks are identical to that illustrated in Figure 2.1. The subproblem for group  $g$ , maximizes plan reduced cost for this group,  $z^g$ , subject to cover, balance and purity constraints.

Maximize:

$$\begin{aligned}
z^g = & \sum_{f \in F} \sum_{i \in L^g} (R_{f,i} - C_{f,i}) x_{f,i} - \sum_{f \in F} \sum_{a \in A^g} \pi_f^{pc} y_{f,a,t_0} \\
& + \sum_f \sum_h \sum_{i \in I(f,h,t)} \pi_{f,h,t}^{bal} x_{f,i} - \sum_f \sum_h \sum_{i \in O(f,h,t)} \pi_{f,h,t}^{bal} x_{f,i} \quad \forall f, h, t
\end{aligned} \tag{3.15}$$

Subject to:

$$\sum_{f \in F} x_{f,i} = 1, \forall i \in L^g \tag{3.16}$$

$$y_{f,a,t^-} + \sum_{i \in I(f,a,t), i \in L^g} x_{f,i} - y_{f,a,t^+} - \sum_{i \in O(f,a,t), i \in L^g} x_{f,i} = 0, \forall f, a \in A^g, t \tag{3.17}$$

$$w_{f,a} \geq x_{f,i} \quad \forall f \in F, a \in A^g, i \in L^g \tag{3.18}$$

$$\sum_{f \in F} w_{f,a} \leq SP_a \quad \forall a \in A^g \tag{3.19}$$

$$x_{f,i} \in \{0,1\}, \forall f \in F, \forall i \in L^g \tag{3.20}$$

$$y_{f,a,t} \geq 0, \forall f, a \in A^g, t \tag{3.21}$$

$$w_{f,a} \in \{0,1\} \quad \forall f \in F, a \in A^g \tag{3.22}$$

The SDMsp is similar to the basic FAM problem for flights in a station group. The first terms in the objective function (Equation 3.15) is to maximize assignment profit. The second term penalizes SDMsp for the use of aircraft in the plan based on the plane count dual from the SDMmp,  $\pi_f^{pc}$ . SDMsp is encouraged to build plans that consume aircraft that are readily available in the network; these fleets have a low value of  $\pi_f^{pc}$ . The third and fourth terms either reward SDMsp for bringing aircraft into the hub or penalize SDMsp for taking aircraft away from the hub. The dual value,  $\pi_f^{bal}$ , indicates the value to SDMmp of having an additional aircraft of fleet  $f$  available at hub  $h$  at time  $t$ . These terms encourage SDMsp to build plans that take advantage of aircraft availability at the hubs and to discourage plans that are infeasible due to aircraft shortages at the hub.



Equations 3.16 are the cover constraints to ensure that each flight in this station group is assigned one aircraft type. Equations 3.17 are the balance constraints for the station within this group. Equations 3.18 and 3.19 are the purity constraints for stations in this group. Equations 3.20, 3.21 and 3.22 ensure that the assignment variables are binary and that the ground arc flow at the stations in this group are non-negative.

### 3.4.7 Generating Plans from SDMsp Solutions

The solution to SDMsp could provide an improving column to SDMmp if the reduced cost is greater than 0. The reduced cost for station group  $g$  is  $RC^g = z^g - \pi_g^{conv}$ . There are two situations in which the reduced cost could be greater than zero. First, this could be a new improving solution; in this case we generate a new column. Second, this solution may duplicate a previous solution already in the SDMmp that is at its upper bound; in this case no new column is required.

If a new column can be generated from the SDMsp solution, the following characteristics of the plan are determined and added to SDMmp:

**Plan Revenue:** 
$$R_p = \sum_{f \in F} \sum_{i \in L^g} R_{f,i} x_{f,i}^p \quad (3.23)$$

**Plan Cost:** 
$$C_p = \sum_{f \in F} \sum_{i \in L^g} C_{f,i} x_{f,i}^p \quad (3.24)$$

**Aircraft Count:** The number of aircraft used in the solution to SDMsp is a characteristic of the plan. For plan  $p$ , station group  $g$ , the plane count is the sum of: 1) aircraft on the ground at stations within this group at the counting time; 2) aircraft assigned to flight legs in the group crossing the counting line:

$$PC_f^p = \sum_{a \in A^g} y_{f,a,t_m} + \sum_{i \in CL(f), i \in L^g} x_{f,i}, \forall f \in F \quad (3.25)$$

**Hub Incidence Vector:** This represents the flow of each fleet type to/from the hubs for plan,  $p$ .

$$q_{f,h,t,p} = \begin{cases} x_{f,i}, & \forall i \in L^g, i \in I(f,h,t) \\ -x_{f,i}, & \forall i \in L^g, i \in O(f,h,t) \\ 0, & \text{otherwise} \end{cases} \quad (3.26)$$

### 3.5 *SDM Model Flow*

In this section we describe the process to solve the linear programming relaxation of the SDM Master Problem, SDMmp. In Sections 3.9 and 3.10 we describe the process to obtain integer solutions. There are four major steps in the SDM solution process:

- Initial Plan Generation
- SDMmp solution
- SDMsp solution, column generation
- Evaluation of stopping criteria

These steps are illustrated in Figure 3.6 and described below.

#### 3.5.1 Initial Plan Generation

For each spoke station group we generate all possible pure plans. That is we generate all plans for each station group that use a single fleet type. In addition, for each group we generate the maximum profit plan by solving SDMsp with all duals variables initially set to 0. While this set of plans does not necessarily guarantee master feasibility it provides a good starting point for the subsequent solution process.

#### 3.5.2 SDMmp Solution

Column generation requires that the master problem is primal feasible. We ensure this by initially relaxing some of the constraints in the master. Artificial variables are added to the plane count and convexity constraints. On the second major iteration, the convexity artificial variables are removed. On this and subsequent iterations, the plane count artificial variable is removed for each fleet satisfying its plane count constraint.

### 3.5.3 SDMsp Solution and Column Generation

On each major iteration, we solve the SDMsp for each station group. The objective function coefficients associated with SDMmp duals are updated and SDMsp is solved. If the optimal objective function value is greater than the SDMmp convexity dual then we check previous solutions to ensure that this is not a duplicate. If it is a new solution then the column is generated as described in Section 3.3.1.7. For some groups, no new columns are generated after the first iterations. Either all columns have been enumerated or the subproblem produces duplicates of previous solutions. In these cases, solving the subproblem on every major iteration is unnecessary. We use a skipping rule. If the subproblem for a station does not produce a new column then skip this station for  $n$  iterations. We typically use  $n=5$ .

The solution to the linear relaxation of SDMsp is generally integer. In some cases, the solutions for some station groups are fractional. There are three options to fractional solutions:

- 1) Convert the assignment variables for this subproblem to integers. This works well for small problems. In the US domestic schedules, this approach consumes a significant amount of memory due to the large number of subproblems.
- 2) Ignore fractional solutions. Fractional solutions occur sporadically, when they do occur we don't generate a new column for this station group. This occasionally leads to infeasibility or poor solution quality.
- 3) Round up the value for the assignment variable with the highest fractional value and resolve the LP relaxation. This approach generates an integer solution on the second pass in every instance in this study. In some cases the quality of the columns is not as high as those produced by the MIP but this process does not consume any more memory than the original subproblem.

### 3.5.4 Stopping Criteria

The process stops if, on a major iteration, no new columns are generated in the subproblems. In some cases, after the first several major iterations, SDM may generate many new columns that provide either a small or no improvement to the objective

function value. We impose an additional stopping criterion based on the maximum reduced cost of the columns being generated,  $RC_{\max} = \max_{g \in G}(RC^g)$ . If  $RC_{\max} \leq \varepsilon$  then STOP. The cutoff is specified as a percentage of the master objective function value. The typical cutoff value is 0.1% of the objective function value. If the model stops and some artificial variables are greater than 0, then the original problem is infeasible.

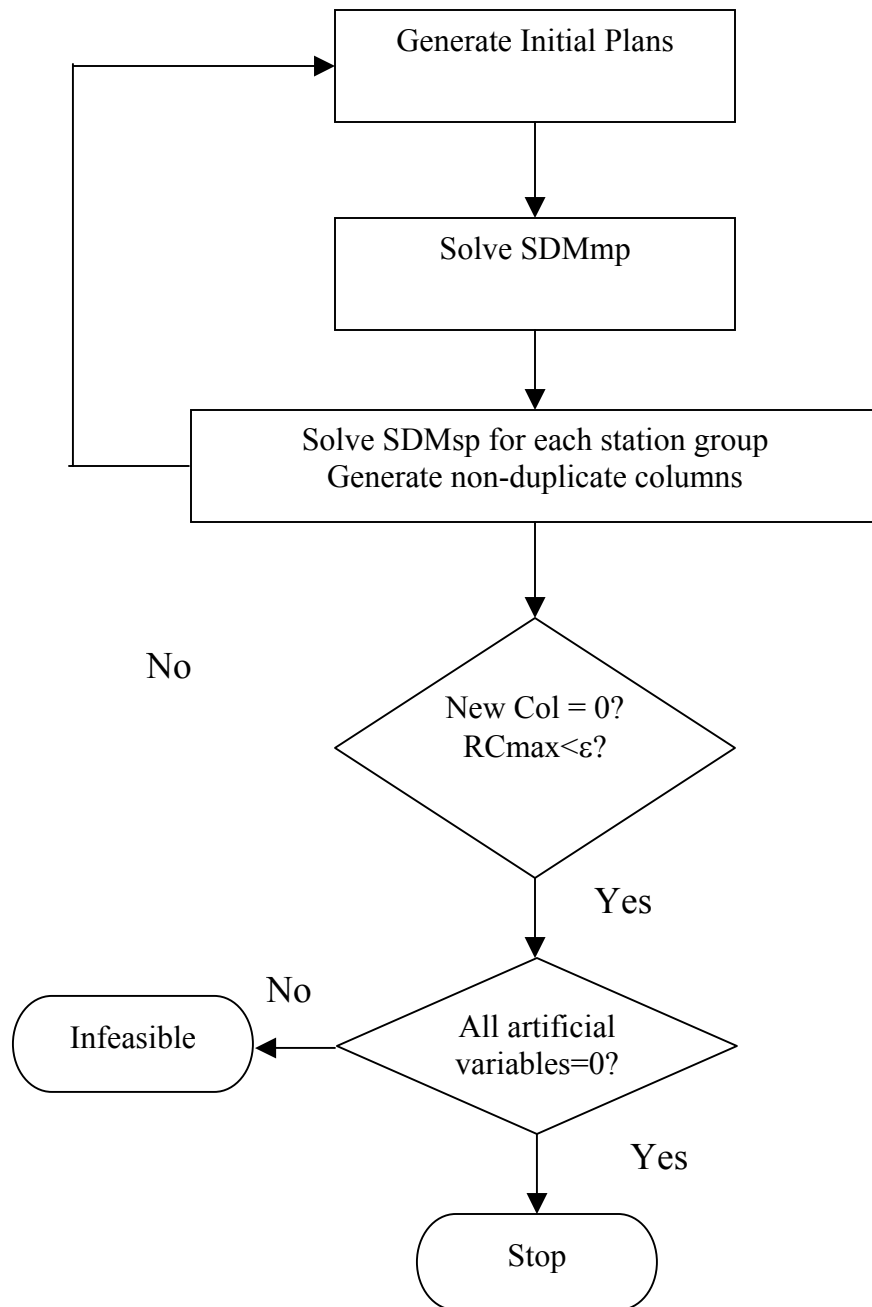


Figure 3.6: SDM model flow.

### 3.6 Hub Selection

In a general network, we need to determine which stations are hubs and which stations are grouped. In order to do this efficiently, we construct the adjacency graph for the network and identify potential hubs based on node degree. When the hubs are removed from the adjacency graph, a valid formulation decomposes into separate components of spokes and groups. We want to balance the size of the master and the size of the largest subproblem. There are two extreme approaches. First, for any network, we can designate all stations as hubs. The resulting master problem is identical to FAM. Second, we can designate one station as a hub and construct groups as necessary to avoid spoke-to-spoke flights. This may result in large subproblems that are too large to solve efficiently. The goal of hub selection is to balance the size of the master and the subproblems.

The adjacency graph,  $G_0$ , has a set of vertices  $V(G_0)$  corresponding to the stations in the schedule and edges  $E(G_0)$ , such that the edge,  $x, y \in V(G_0)$  if the schedule contains at least one flight from station  $x$  to station  $y$ . Let  $d(v)$  be the degree of vertex  $v$  in  $G_0$ . We identify a candidate hub based on maximum degree in the adjacency graph. Let  $H$  be a set of candidate hubs; the first candidate hub,  $h_1$  is the vertex with maximum degree in the initial graph,  $G_0$ . This vertex and the edges incident to it are removed, to form a new subgraph  $G_1$ . This process continues until all vertices have been removed. The  $i$ th candidate hub is found as:  $h_i = \underset{j}{\operatorname{argMax}}\{d(v_j), v_j \in V(G_{i-1})\}$ . The next subgraph,  $G_{i+1}$  has a reduced set of vertices,  $V(G_{i+1}) = (V(G_i) \setminus h_i)$ . The hubs are removed and the process continues in the resulting subgraph until all stations are identified as hubs. This results in an ordering of candidate hubs. We can choose the number of hubs to use that results in an appropriate balance between master and subproblem sizes. This process is illustrated in Figures 3.7, 3.8 and 3.9.

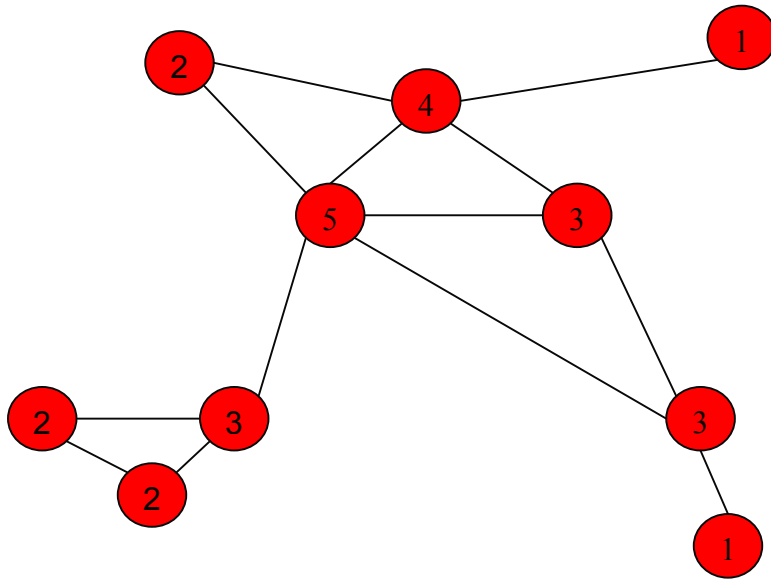


Figure 3.7: Initial adjacency graph,  $G_0$  with node degree.

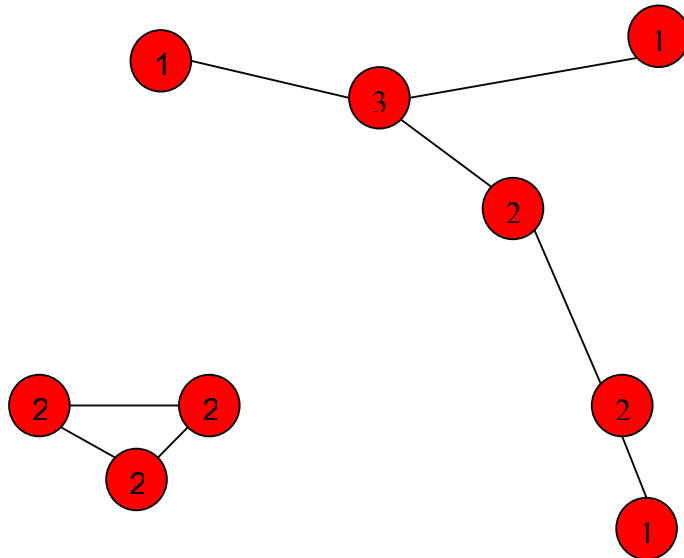


Figure 3.8: Adjacency graph,  $G_l$ , after deleting first candidate hub,  $h_l$ .

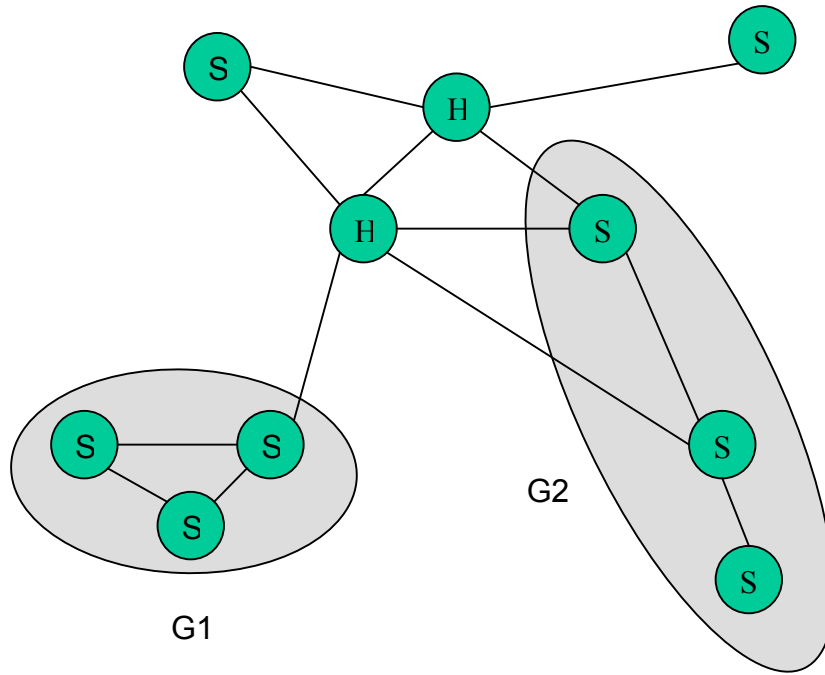


Figure 3.9: Initial adjacency graph,  $G_0$ , with hubs and station groups identified.

Table 3.3 illustrates this process in the US domestic schedule. If one station is designated as a hub, then there are 38 components (spokes and groups). The maximum group contains 170 stations (out of 209) and 3176 flights (out of 4182). With 31 hubs, the schedule decomposes into 179 individual stations with a maximum size of 25 daily operations. The best balance appears to be achieved with 11 hubs. Additional hubs increase master size, but do not reduce maximum group size significantly. Table 3.4 illustrates the same process on the international weekly schedule.



Table 3.3: Hub selection for the US daily schedule.

Hubs	Groups	Max Group Size		H2H Flights
		Stations	Flights	
1	38	170	3176	0
2	77	130	2312	17
3	106	84	1793	42
4	137	55	1171	88
5	141	48	992	119
6	150	39	770	173
7	158	30	620	235
8	167	19	358	261
9	172	9	222	309
10	175	4	108	375
11	176	4	68	435
12	177	4	68	441
31	179	1	25	844

Table 3.4: Hub selection for the international weekly schedule.

Hubs	Groups	Max Group Size		H2H Flights
		Stations	Flights	
1	14	34	1409	0
2	18	29	1218	71
3	20	25	1074	134
4	21	20	735	440
5	25	10	523	506
6	26	10	301	772
7	31	4	186	856
8	32	4	113	912
9	34	2	70	940
16	29	1	52	1186

As the number of hubs increases, the number of flights assigned in the subproblems is reduced and the number of flights assigned in the master increases. Table 3.5 summarizes the impact on the number of flights in the plans and H2H flights (not in the plans) and SDM Master problem size.

Table 3.5: Impact of hub selection on SDMmp size.

Case	Hubs	Flts in Plans	H2H Flts	Initial Master Size	
				Rows	Cols
Int	6	1586	772	3540	5780
	7	1502	856	3700	5996
	8	1446	912	3903	6339
	9	1418	940	3961	6397
	16	1158	1200	4486	7130
US	7	3947	235	4085	5669
	10	3807	375	4602	7109
	11	3747	435	4816	7768
	20	3567	615	5568	9740
	31	3338	844	6537	12439

There are three hub configurations that we test:

- Min SDMsp: Minimum number of hubs to achieve relatively small sub-problems. For example in the US domestic case the size of the subproblems drops significantly with 11 hubs.
- Disconnect: This is the minimum number of hubs so that the largest station group size is 1. In the US domestic case this is 31 hubs.
- Non\_SP1: Disconnecting hubs plus spokes without pure station constraint (SP1). We set up subproblems only for stations with a hard purity constraint. In the US domestic case this is 39 hubs.

Table 3.6 summarizes the configurations for each schedule.

Table 3.6: Hub configurations for each schedule.

	Min SDMsp	Disconnect	Non-SP1
Star	3	1	27
Int	6	16	26
US	11	31	39

### **3.7 SDM Computational Results**

The performance of SDM with various hub configurations is summarized in Table 3.7. In the Star7 case, the solution efficiency and quality improves with a greater number of hubs. In the Base case, the number of major iterations drops from 100 to 3 when hubs are increased from 1 to 27. The number of plans required drops from 1613 to 26. Total CPU time drops from 55 to 7 seconds while the profit increases from \$65.35 to \$65.38 million. This result is surprising since we anticipated that SDM would work best in a star network. There are several relatively large spokes in Star7 network and there is a large number of possible fleeting solutions for them. The quality of the solutions increases slowly and this process requires many major iterations. By including large spokes in the master, the number of possible fleeting solutions at the remaining spokes is reduced and the number of major iterations is reduced as well. SDM performance in the Base case is comparable to FAM. In the Maximum purity case, there are 3 stations that do not have purity. By making these stations hubs, we can enumerate all possible fleeting plans. Generating the columns and solving the master require 5% of the FAM solution time. With additional hubs, the master time increases to nearly equal the 1-hub timing. In the Moderate case, fewer spokes have absolute purity; more spokes are allowed 2 fleet types. The best option again is 27 hubs; the performance is close to FAM.

In the Int7 schedule, we have more actual hubs and more hub-to-hub flights. This configuration makes SDM much less efficient. In the Base and Moderate cases SDM requires 19 and 41 major iterations; as a result, performance is significantly worse than FAM. In the Maximum purity case, we can enumerate plans in the 26-hub configuration and performance is significantly better than FAM.

In the US7 and US19 cases, SDM performance is poor in the Base case. In the Maximum and Moderate purity cases, SDM provides significant benefits. The time required to solve the Maximum purity cases is 8% and 1% of the FAM time for US7 and US19 respectively.

These results indicate that a concentrated hub and spoke network by itself does not guarantee that SDM will improve performance. A hub and spoke network combined with difficult constraints imposed at the station level does provide an environment in which SDM can significantly outperform FAM.

Table 3.7: SDM performance for various network configurations.

Schedule	Purity	Hubs	Major Its	Plans	Profit \$mm	CPU seconds			
						Master	ColGen	Total	SDM/FAM
Star7	Base	1	100	1613	65.35	51.17	4.19	55.36	878.73%
		3	27	531	65.37	22.58	0.97	23.55	373.81%
		27	3	26	65.38	6.83	0.19	7.02	111.43%
	Max	1	200	488	64.08	37.31	4.36	41.67	175.08%
		3	1	82	64.18	1.17	0.15	1.32	5.55%
		27	2	22	64.18	22.72	0.05	22.77	95.67%
	Mod	1	100	1434	65.29	50.44	13.42	63.86	473.04%
		3	26	477	65.31	20.72	1.86	22.58	167.26%
		27	2	22	65.32	15.59	0.03	15.62	115.70%
Int7	Base	6	38	399	82.55	184.39	2.46	186.85	1478.24%
		16	21	315	82.55	98.56	0.90	99.46	786.87%
		26	13	169	82.55	57.42	0.44	57.86	457.75%
	Max	6	10	60	81.43	29.08	6.01	35.09	77.29%
		16	11	73	81.43	25.67	0.36	26.03	57.33%
		26	1	42	81.43	12.70	0.10	12.80	28.19%
	Mod	6	23	250	82.41	96.91	24.87	121.78	385.87%
		16	18	266	82.42	89.58	1.47	91.05	288.50%
		26	15	171	82.42	71.16	0.81	71.97	228.04%
US7	Base	11	13	2285	17.51	31.94	8.16	40.10	338.97%
		31	6	1717	17.51	17.38	5.88	23.26	196.62%
		39	6	1650	17.51	21.84	5.60	27.44	231.95%
	Max	11	18	1045	14.67	11.19	4.29	15.48	8.39%
		31	19	1061	14.67	45.97	3.00	48.97	26.53%
		39	6	1016	14.67	19.25	2.49	21.74	11.78%
	Mod	11	9	1353	16.66	9.22	9.06	18.28	20.04%
		31	9	1311	16.65	38.99	2.38	41.37	45.35%
		39	7	1237	16.66	45.27	5.47	50.74	55.62%
US19	Base	11	29	3412	19.36	627.75	21.91	649.66	256.78%
		31	15	1946	19.36	555.98	9.94	565.92	223.68%
		39	13	1608	19.36	701.77	7.26	709.03	280.25%
	Max	11	23	2602	15.84	85.07	33.16	118.23	1.42%
		31	30	2641	15.83	496.95	17.72	514.67	6.20%
		39	1	2562	15.84	108.17	15.67	123.84	1.49%
	Mod	11	47	2564	18.16	189.84	220.70	410.54	20.33%
		31	46	2360	18.13	1168.70	87.10	1255.80	62.18%
		39	9	1999	18.13	807.97	52.19	860.16	42.59%

### **3.8 Dual Improvement Approach to SDM**

In a general network, the SDM approach is less efficient than FAM in the base case due to the large number of major iterations required to produce a good solution. The quality of the columns is limited by the quality of the dual solution from the master, which is in turn affected by the quality of the columns. As the model approaches the optimal solution the rate of improvement slows significantly.

Approaches to improving column generation efficiency include dual stabilization, a combination of perturbation and penalties that can reduce the number of major iterations required to produce good solutions (du Merle et al. 1999). Dual stabilization has been shown to improve the efficiency of the aircraft routing problem (du Merle and Vial 2003). Lemarechal develops a method of identifying a new search direction based on a bundle of previous subgradients. The new direction is a convex combination where the weight on each previous subgradient is based on accuracy at the current solution (Hiriart-Urruty and Lemarechal 1993). Thengvall et al. (2003) use bundle methods for aircraft schedule recovery. Lagrangian/surrogate methods have also proven beneficial in column generation (Senne and Lorena 2000). Hu and Johnson (1999) develop a primal-dual approach in which subproblem solutions are used to improve the duals, while maintaining dual feasibility.

In this section we investigate a method of improving the quality of the dual solution on each major iteration. Our approach is to find the optimal dual solution in a convex region bounded by previous dual solutions. This approach should improve the quality of the duals and should reduce the number of major iterations required to produce good solutions.

We start with the SDM master formulation described in Section 3.1. The SDM master problem formulation notation is simplified to facilitate the discussion:

$C_H$  : vector of objective function coefficients for H2H flight legs (revenue minus cost).

$X_H$  : vector of decision variables associated with H2H flight assignments and hub ground flows.

$P_p = R_p - C_p$  : profit associated with plan  $p$ .

$A_H$  : matrix of coefficients for H2H cover, hub balance and plane count constraints.

$A^p$  : vector of coefficients for plane count and hub incidence for plan  $p$ .

$b$  : vector of right hand side values associated with plane count and hub balance constraints.

The SDM master problem is:

$$\text{Maximize:} \quad C_H X_H + \sum_{g \in G} \sum_{p \in P^g} P_p x_p \quad (3.27)$$

Subject to:

$$\sum_{p \in P^g} x_p = 1, \forall g \in G \quad (3.28)$$

$$A_H X_H + \sum_{g \in G} \sum_{p \in P^g} A^p x_p = b \quad (3.29)$$

$$X_H \geq 0, x_p \geq 0, \forall p \in P \quad (3.30)$$

The objective function has separate terms for the profit of the H2H flights (first term) versus the plan flights (second term). Equations 3.28 are the convexity constraints for each station group. Equations 3.29 are the H2H cover, hub balance and plane count constraints. Equations 3.30 ensure non-negativity. The dual variables for Equations 3.28 are  $\pi_g^{conv}$ ; for Equations 3.29 the duals are  $\pi$ .

The dual of the SDMmp is to minimize:

$$\pi b + \sum_{g \in G} \pi_g^{conv} \quad (3.31)$$

Subject to:

$$\pi A_H \geq C_H \quad (3.32)$$

$$\pi_g^{conv} + \pi A^p \geq P_p, \forall g \in G, p \in P^g \quad (3.33)$$

Minimizing the SDMmp dual for a given  $\pi$  is equivalent to:

$$\min \left\{ \sum_{g \in G} \pi_g^{conv} \mid \pi_g^{conv} \geq P_p - \pi A^p, \forall g \in G, p \in P^g \right\} \quad (3.34)$$

or from duality:

$$\max \left\{ \sum_{g \in G} \sum_{p \in P^g} (P_p - \pi A^p) x_p \mid \sum_{p \in P^g} x_p = 1, \forall g \in G, x_p \geq 0, \forall p \in P \right\} \quad (3.35)$$

Substituting 3.35 into the SDM dual yields the dual lagrangian,  $L(\pi)$ :

$$\min L(\pi) = \min \pi b + \max \left\{ \sum_{g \in G} \sum_{p \in P^g} (P_p - \pi A^p) x_p \mid \sum_{p \in P^g} x_p = 1, \forall g \in G, x_p \geq 0, \forall p \in P \right\} \quad (3.36)$$

or,

$$\min L(\pi) = \min \pi b + \sum_{g \in G} S^g(\pi) \quad (3.37)$$

where,  $S^g(\pi)$  is the optimal solution value to the SDM subproblem for station group  $g$ , given SDM master dual solution  $\pi$ .

In order to simplify the problem, and to ensure dual feasibility we restrict the minimization of  $L(\pi)$  to a convex region bounded by previous dual solutions:  $\pi^1 \dots \pi^t$ .

Let  $\lambda$  be a vector of weights on these solutions, and let  $\pi^\lambda$  be the convex combination of the previous solutions based on these weights. The restricted lagrangian minimization becomes:

$$\min_{\{\pi^1, \dots, \pi^t\}} L(\pi^\lambda) = \min \left\{ \pi^\lambda b + \sum_s S^s(\pi^\lambda) \mid \pi^\lambda = \sum_j \lambda_j \pi^j, \sum_j \lambda_j = 1, \lambda_j > 0 \right\} \quad (3.38)$$

Equation 3.38 is equivalent to:



$$\min_{\{\pi^1, \dots, \pi^t\}} L(\pi^\lambda) = \min \left\{ \begin{array}{l} \pi^\lambda b + \sum_{g \in G} \pi_g^{conv} \mid \\ \pi_g^{conv} \geq P_p - \pi^\lambda A^p \forall p \in P^g \\ \pi^\lambda = \sum_{j=1..t} \lambda_j \pi^j, \sum_{j=1..t} \lambda_j = 1, \lambda_{1..t} > 0 \end{array} \right\} \quad (3.39)$$

or:

$$\min_{\{\pi^1, \dots, \pi^t\}} L(\pi^\lambda) = \min \left\{ \begin{array}{l} \sum_{j=1..t} \lambda_j \pi^j b + \sum_{g \in G} \pi_g^{conv} \mid \\ \sum_{j=1..t} \lambda_j \pi^j A^p + \pi_g^{conv} \geq P_p, \forall p \in P^g \\ \sum_{j=1..t} \lambda_j = 1, \lambda_{1..t} \geq 0 \end{array} \right\} \quad (3.40)$$

The dual of 3.40 provides a method to minimize the lagrangian in terms of the primal SDMmp. We refer to the dual of Equation 3.40 as the Second Stage Master Problem, SSMP:

Maximize:

$$z_0 + \sum_{p \in P} P_p x_p \quad (3.41)$$

Subject to:

$$\sum_{p \in P^g} x_p = 1, \forall g \in G \quad (3.42)$$

$$z_0 + \sum_{p \in P} \pi^j A^p x_p = \pi^j b, \forall j = \{1..t\} \quad (3.43)$$

$$x_p \geq 0 \quad (3.44)$$

Where,  $z_0$  is the dual of the convexity constraint on  $\lambda$  in Equations 3.40. Given an optimal solution to the SSMP, the improved dual  $\pi^\lambda$  is found as:

$$\pi^\lambda = \sum_{j=1..t} \lambda_j \pi^j \quad (3.45)$$

where  $\lambda_j$  is the dual on the  $j^{\text{th}}$  SSMP constraint in Equations 3.40.

### 3.8.1 Solution Process

The dual improvement approach introduces a minor iteration loop into the solution process. A major iteration involves solving the SDM master problem. A minor iteration begins by using the SDMmp dual solution to construct a new row in the second stage master problem (SSMP). The SSMP produces an improved dual solution  $\pi^\lambda$ , in the convex region bounded by previous dual solutions,  $\pi^\lambda$  is used to generate new columns. These columns are added back to the SSMP, and it is solved again. This minor loop (SSMP – Column Generation) runs until no new columns are generated. At this point we begin the next major iteration by adding all the new columns to the master, and it is solved again. Figure 3.10 illustrates this process on the nth major iteration.

The impact of the dual improvement process on  $L(\pi)$  is illustrated in Figure 3.11. Suppose we have 2 plans  $p_0$  and  $p_1$ , they define the dual feasible region. Further suppose we have two previous dual solutions ( $\pi_1$  and  $\pi_2$ ). On the first minor iteration, before any new columns are generated, the most recent solution  $\pi_2$  is optimal for the SSMP; the first weighted dual  $\pi^\lambda$  sent to the column generation process will equal the most recent master dual solution,  $\pi_2$ .

For example, when plan  $p_2$  is generated, the optimal dual solution moves from  $\pi_2$  to  $\pi_{2.1}$  (Figure 3.12). On the next minor iteration a plan  $p_3$  is generated based on  $\pi_{2.1}$ ; this cuts off the previous optimal solution and moves the optimal dual solution to  $\pi_{2.2}$ . (Figure 3.13). This process of improving the dual solution continues until no new plans are generated that cut-off the current optimal solution. The minor iteration stops and the next major iteration begins.

Each new plan is a cut in the dual problem and has the following impacts:

- Improving plans cut-off the current dual solution
- The optimal solution changes from  $\pi_{m,n}$  to  $\pi_{m,n+1}$ , where m=major iteration number, n=minor iteration number

- The objective function increases from  $L(\pi_{m,n})$  to  $L(\pi_{m,n+1})$  where  $L(\pi_{m,n+1}) > L(\pi_{m,n})$ .

We can improve the dual solution significantly without resolving the master. The benefit of this approach is that the SSMP is much smaller than the master problem. The number of rows in the SSMP is equal to the number of previous dual solutions plus the convexity constraint. In the SDM master there is a row for each hub departure and arrival node, and flight leg:  $|L||F|+|L|$ . The number of columns is equal to the number of ground arcs plus the number of plans (Table 3.8).

Table 3.8: SDMmp vs SSMP Problem Sizes.

	SDM Master	SSMP
Rows	$ L  F + P $	$ \pi +1$
Columns	$ L  F + P $	$ P +1$

The SSMP optimizes the plans, assuming the non-plan assignment variables (hub to hub assignments and ground network flow variables at the hubs) are fixed. The SSMP solutions are dual feasible but not necessarily primal feasible to the master. The total number of iterations will increase using SSMP, many of these iterations will be minor rather than major. This should decrease the total number of major iterations and time required to solve the SDM problem.

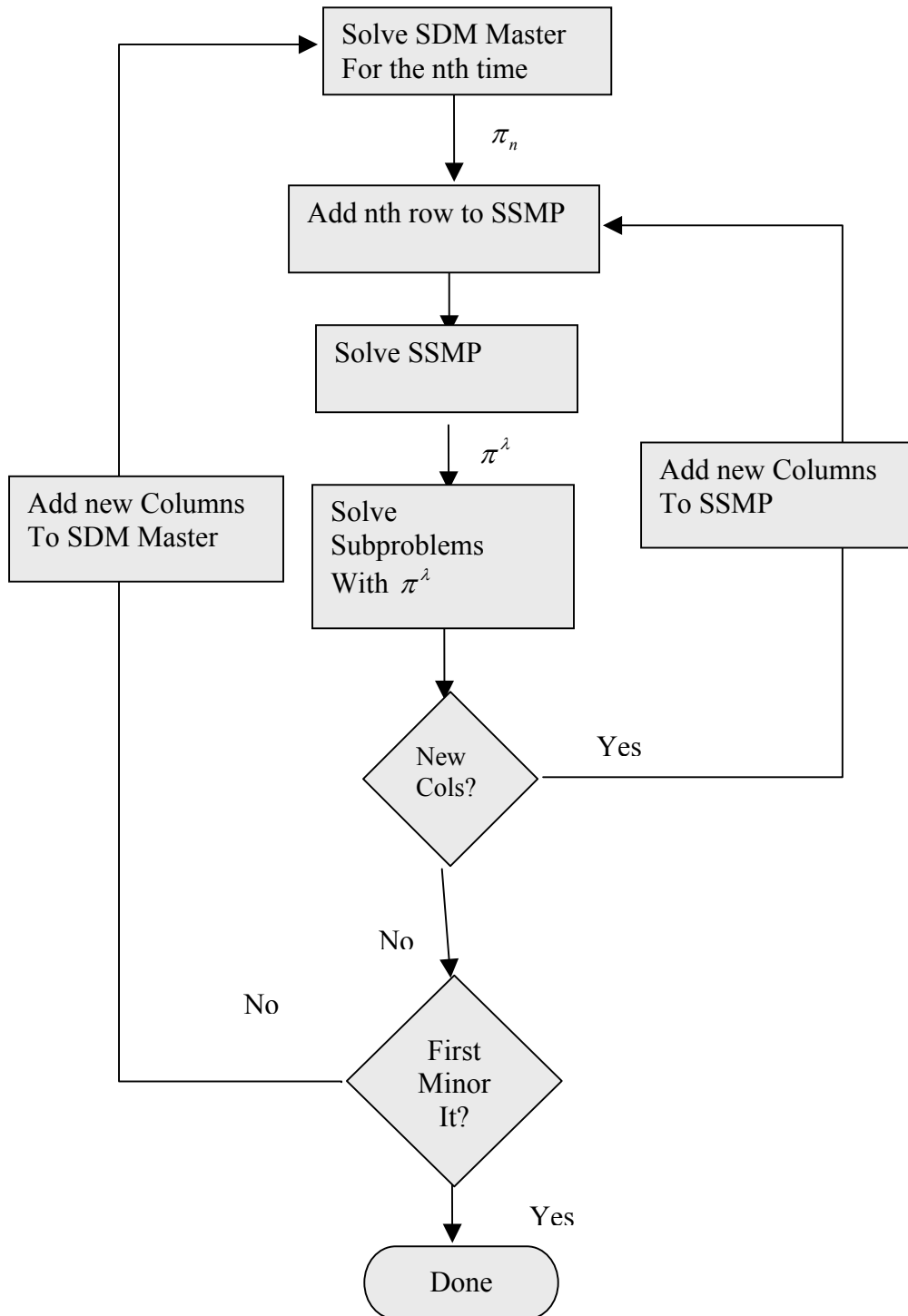


Figure 3.10: The Dual Improvement Algorithm on the  $n$ th major iteration.

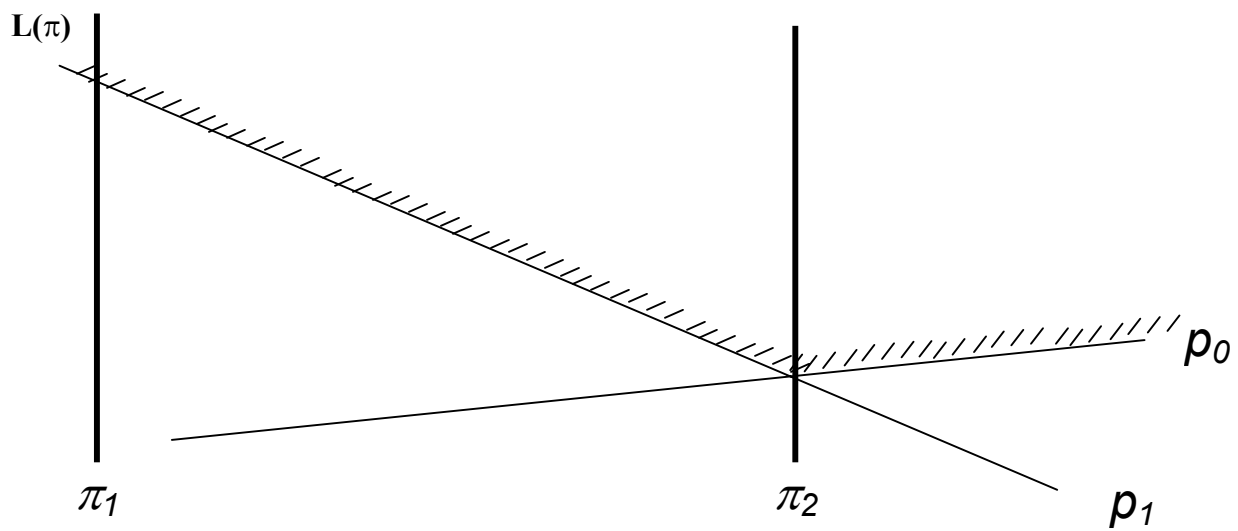


Figure 3.11:  $L(\pi)$  after major iteration number 2.

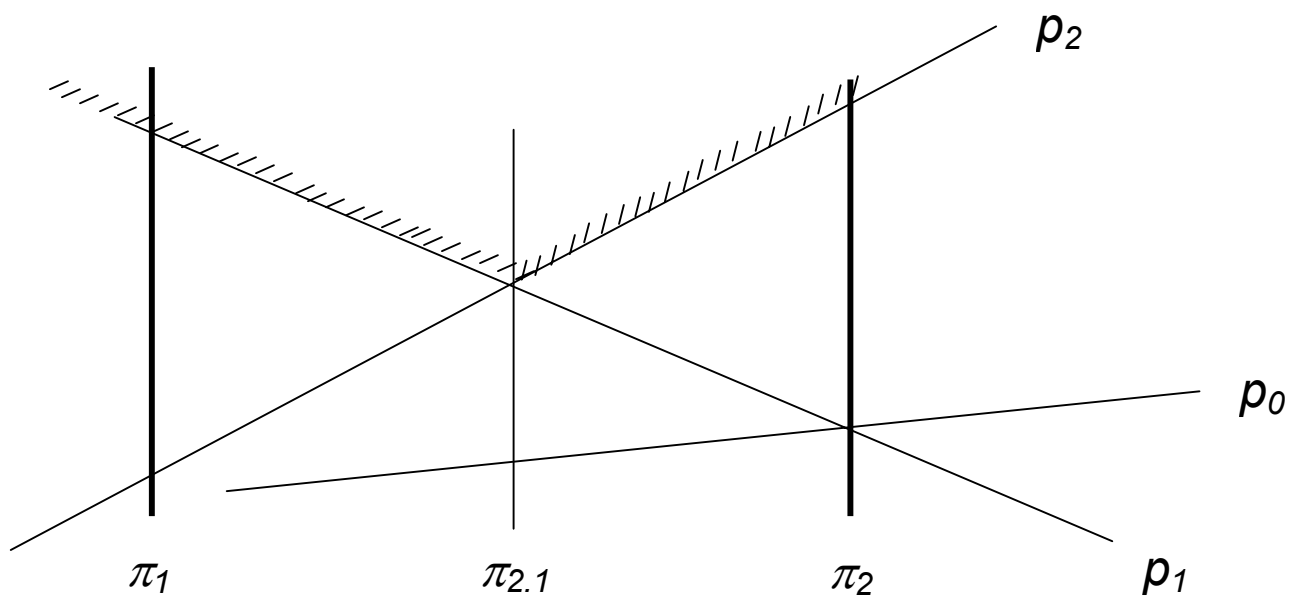


Figure 3.12:  $L(\pi)$  after iteration number 2.1.

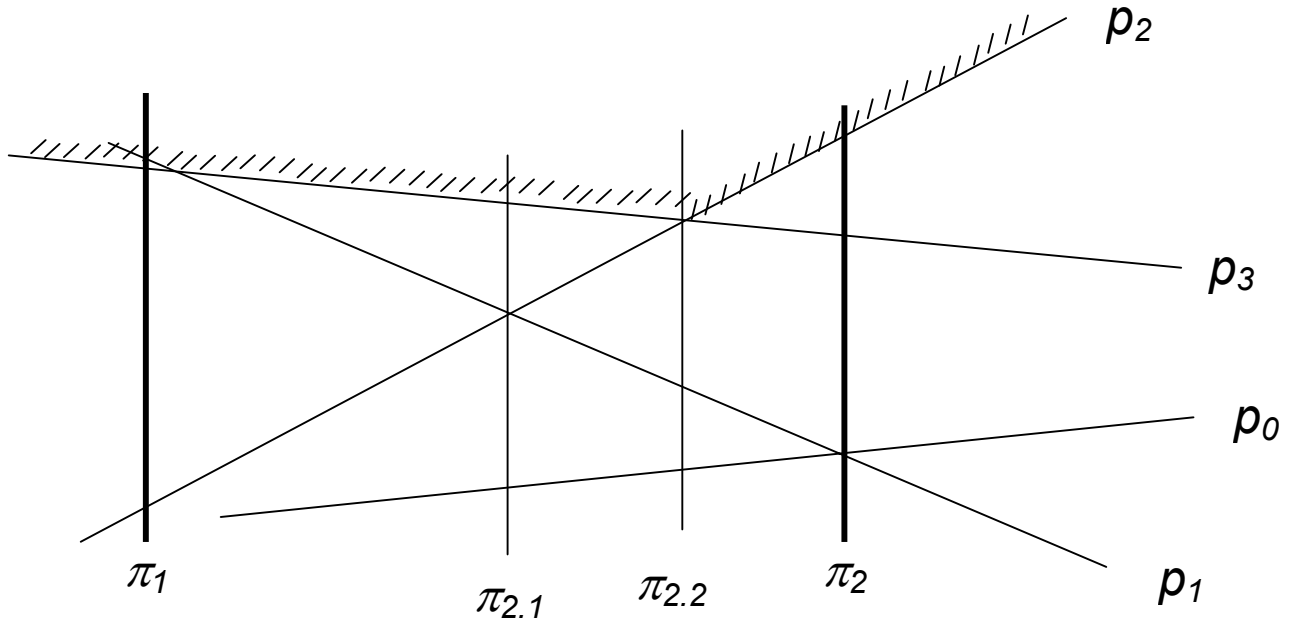


Figure 3.13:  $L(\pi)$  after iteration number 2.2.

On each major iteration, the dual solution improves and the initial solutions become less valuable in the SSMP. We do not necessarily need to use all previous dual solutions in the SSMP. There are 3 possible methods to implement the dual improvement algorithm:

1. Use all previous duals, as described previously. The dimensionality of the convex region in which the SSMP optimizes increases on each major iteration.
2. Use at most  $k$  previous dual solutions. When  $k$  is reached, drop the previous solution with minimum weight, and replace it with the next SDM master dual solution.
3. Use 2 previous dual solutions, the most recent SDM master dual and the final  $\pi^\lambda$  from the last major iteration.

Figure 3.14 illustrates the first 3 major iterations using Option 1. On iteration 1 there is only one dual solution, it is the optimal solution for SSMP. On iteration 2, SSMP

optimizes on the line between  $\pi_1$  and  $\pi_2$ . The optimal solutions are  $\pi_{2.1}$  and  $\pi_{2.2}$ . On the third iteration, SSMP optimizes in the region defined by  $\pi_1$ ,  $\pi_2$  and  $\pi_3$ . The optimal solutions may be interior to the region,  $\pi_{3.1}$  or on the boundary,  $\pi_{3.2}$ .

Figure 3.15 illustrates the process for Option 3. In this case the SSMP finds an improving dual by conducting a line search between the new dual solution and the previous weighted dual solution. Starting with  $\pi_1$  and  $\pi_2$ , the SSMP solution on the first minor iteration produces a weighted dual of  $\pi_{2.1}$  new plans are generated and the second minor iteration results in  $\pi_{2.2}$ . Assume that no new plans are generated using  $\pi_{2.2}$  and the minor loop ends at this point. Major iteration 3 begins by adding the new columns to the SDM master and then the master is resolved. The new dual solution is  $\pi_3$ . The SSMP model is updated with a new row associated with  $\pi_3$  and a row for the last weighted dual  $\pi_{2.2}$ . The subsequent minor iterations find improving solutions on the line between  $\pi_{2.2}$  and  $\pi_3$ .

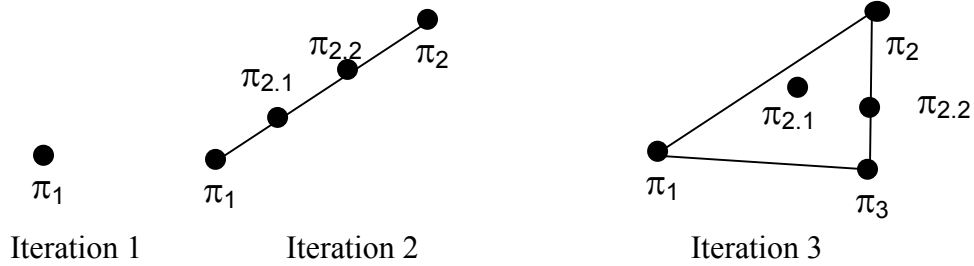


Figure 3.14: SSMP with Option 1, using all previous duals.

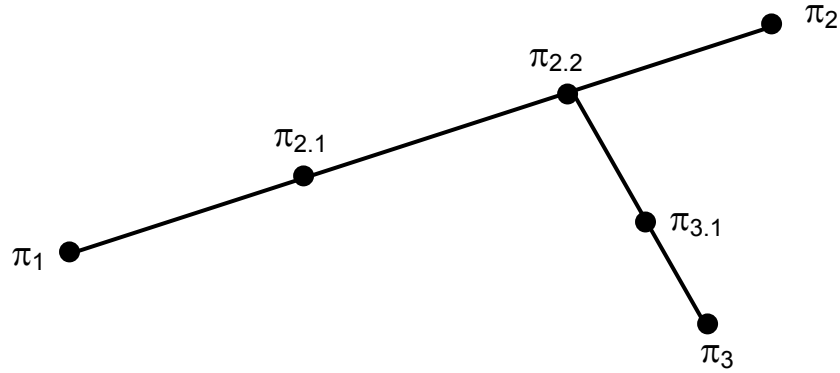


Figure 3.15: SSMP with Option 3, using previous convex combination dual.

### 3.8.2 Computational Results

The dual improvement approach is run for all schedules and purity scenarios. In each case we run the SSMP using Option 3, Option 2 with 3 previous dual solutions and Option 1 with all previous dual solutions. Table 3.9 summarizes the results, it contains the number of major and minor iterations, the number of plans generated, the profit (\$ millions) and the CPU time required for the master problem, the column generation sub-problems and the SSMP. Finally we compare the total CPU time for the dual improvement approach versus FAM. Note that the dual improvement approach is not applied to Star7. Since the number of major iterations required by SDM is low, there is not room for improvements from SSMP. In all other cases, the dual improvement approach reduced the number of major iterations (except for US7 with no purity constraints). The reduction using Option 3 is up to 50%; when all previous dual solutions are used (Option 1) the reduction is up to 70% (US19 Moderate purity). The dual improvement approach requires up to 112 minor iterations (US19 Max). But since the SSMP typically requires 0.01 seconds to solve, the minor iterations add a relatively small amount of additional CPU time. Increasing the number of previous duals used typically reduces the number of major iterations. There is one notable exception. In the Int7 Moderate scenario, using all previous dual solutions increases the number of major



iterations from 15 to 18. This is a result of the stopping criterion used; the reduced cost of the new columns generated continues to remain relatively high even though the solution is close to optimal. In most cases using Option 3 or Option 2 with 3 previous solutions provides all the benefits of this approach. In only one case (US19 Mod) using all previous solutions yields the best results. The total CPU time required to solve FAM using the dual improvement approach is reduced by up to 40% versus the single model. The biggest reductions occur in cases where SDM requires many major iterations, and is relatively inefficient compared to FAM.

Table 3.9: Summary of SSMP results.

Schedule	Purity	Hubs	SSMP Option	Its		Plans	Profit	CPU				
				Major	Minor					Master	ColGen	SSMP
Star7	Base	27		3		26	65.38	6.83	0.19		7.02	111.43%
	Max	3		1		82	64.18	1.17	0.15		1.32	5.55%
	Mod	27		2		22	65.32	15.59	0.03		15.62	115.70%
Int7	Base	27		13		169	82.55	57.42	0.44		57.86	457.75%
		27	3	9	25	231	82.55	50.42	0.67	0.03	51.12	404.43%
		27	2, 3sols	9	30	238	82.55	49.45	0.66	0.02	50.13	396.60%
		27	1	8	26	227	82.55	42.17	0.62	0.03	42.82	338.77%
	Max	27		1		42	81.43	12.7	0.1		12.8	28.19%
	Mod	27		15		171	82.42	71.16	0.81		71.97	228.04%
		27	3	9	25	200	82.42	43.77	1.05	0.04	44.86	142.14%
		27	2, 3sols	11	48	246	82.42	52.36	1.42	0.16	53.94	170.91%
		27	1	18	67	339	82.42	105.16	2.17	0.42	107.75	341.41%
	US7	Base	31		6		1717	17.51	17.38	5.88		23.26
31			3	6	11	1843	17.52	17.59	6.79	0.13	24.51	207.19%
31			2, 3sols	6	12	1848	17.52	17.59	6.93	0.16	24.68	208.62%
Max		11		18		1045	14.67	11.19	4.29		15.48	8.39%
		11	3	10	32	1066	14.67	9.92	4.95	0.13	15	8.13%
		11	2, 3sols	9	36	1072	14.68	9.47	4.97	0.16	14.6	7.91%
Mod		11		9		1353	16.66	9.22	9.06		18.28	20.04%
		11	3	7	14	1378	16.66	7.47	9.97	0.09	17.53	19.22%
		11	2, 3sols	7	14	1381	16.66	7.84	9.85	0.09	17.78	19.49%
US19	None	31		15		1946	19.36	555.98	9.94		565.92	223.68%
		31	3	11	37	2646	19.36	376.97	13.89	0.79	391.65	154.80%
		31	2, 3sols	11	41	2747	19.36	371.56	13.45	1.19	386.20	152.65%
		31	1	11	36	1978	19.35	382.25	14.04	2.93	399.22	157.79%
	Max	11		23		2602	15.84	85.07	33.16		118.23	1.42%
		11	3	12	57	2666	15.84	89.44	33.01	0.42	122.87	1.48%
		11	2, 3sols	10	62	2671	15.84	66.44	30.69	0.48	97.61	1.18%
		11	1	10	112	2733	15.84	73.33	36.39	1.02	110.74	1.33%
	Mod	11		47		2564	18.16	189.84	220.70		410.54	20.33%
		11	3	24	102	2758	18.16	174.63	214.86	1.69	391.18	19.37%
		11	2, 3sols	20	89	2767	18.16	142.02	226.09	1.55	369.66	18.30%
		11	1	15	59	2617	18.16	107.73	227.40	4.98	340.11	16.84%

Table 3.10 contains the details for one scenario (US19 Moderate Purity, using Option 3).

For each iteration, Table 3.10 contains, the SSMP objective function value  $L(\pi)$ , the weights on the previous and current duals, and the number of new columns generated. In this case the first 2 major iterations produced infeasible solutions, plane count and spoke

coverage were violated. On the first minor iteration of each major iteration,  $L(\pi)$ , is equal to the current objective function of the master problem, the weight on the current dual is 1. Columns generated at this point are identical to those in SDM. On the subsequent minor iterations, some weight shifts to the previous dual, and the SSMP objective function,  $L(\pi)$ , increases. The amount of weight change and the number of new columns generated in the minor iterations reduces until the weights converge and no new columns are generated. This ends the major iteration. In this scenario, there are typically 4-5 minor iterations per major iteration.

The  $L(\pi)$  provides bounds on the optimal solution. The 0<sup>th</sup> minor iteration provides a lower bound, and is equal to the current SDM master solution. On the final minor iteration,  $L(\pi)$  provides an upper bound that is dual feasible, but primal infeasible. As illustrated in Figure 3.16 the gap between these bounds drops on every major iteration. This can be used as an improved stopping criterion.

Compared to SDM, the quality of the dual improvement solution is significantly higher after the same number of major iterations. The optimality gap of SDM and SSMP versus iteration is summarized in Figure 3.17. For this we can see that the optimality gap is reduced by almost 50% in iterations 3 and 8 using 2 previous dual solutions. The use of additional dual solutions provides a small additional reduction in optimality gap.

Table 3.10: SSMP details for US19 moderate purity.

Iteration		L( $\Pi$ )	Weights		New Columns
Major	Minor		Previous Dual	Current Dual	
0		INF		1.000	73
1	0	INF	0.000	1.000	68
1	1	INF	0.520	0.480	57
1	2	INF	0.143	0.857	37
1	3	INF	0.143	0.857	2
1	4	INF	0.143	0.857	0
2	0	\$16,780,500	0.000	1.000	67
2	1	\$20,581,500	0.000	1.000	0
3	0	\$17,007,400	0.000	1.000	58
3	1	\$18,597,300	0.352	0.648	46
3	2	\$18,954,400	0.223	0.777	18
3	3	\$18,980,400	0.220	0.780	1
3	4	\$18,980,500	0.220	0.780	0
4	0	\$17,565,400	0.000	1.000	63
4	1	\$18,444,000	0.494	0.506	52
4	2	\$18,692,900	0.358	0.642	20
4	3	\$18,706,100	0.416	0.584	5
4	4	\$18,706,500	0.416	0.584	0
5	0	\$17,872,400	0.000	1.000	56
5	1	\$18,225,200	0.421	0.579	39
5	2	\$18,355,400	0.567	0.433	21
5	3	\$18,368,700	0.488	0.512	7
5	4	\$18,370,000	0.513	0.487	1
5	5	\$18,370,200	0.509	0.491	1
5	6	\$18,370,200	0.509	0.491	0
6	0	\$18,066,500	0.000	1.000	57
6	1	\$18,215,000	0.333	0.667	29
6	2	\$18,230,700	0.207	0.793	9
6	3	\$18,232,800	0.216	0.784	1
6	4	\$18,232,800	0.216	0.784	0
7	0	\$18,124,000	0.000	1.000	40
7	1	\$18,175,400	0.297	0.703	21
7	2	\$18,180,600	0.274	0.726	1
7	3	\$18,180,600	0.274	0.726	0
8	0	\$18,139,300	0.000	1.000	30
8	1	\$18,166,800	0.508	0.492	14
8	2	\$18,173,400	0.504	0.496	0

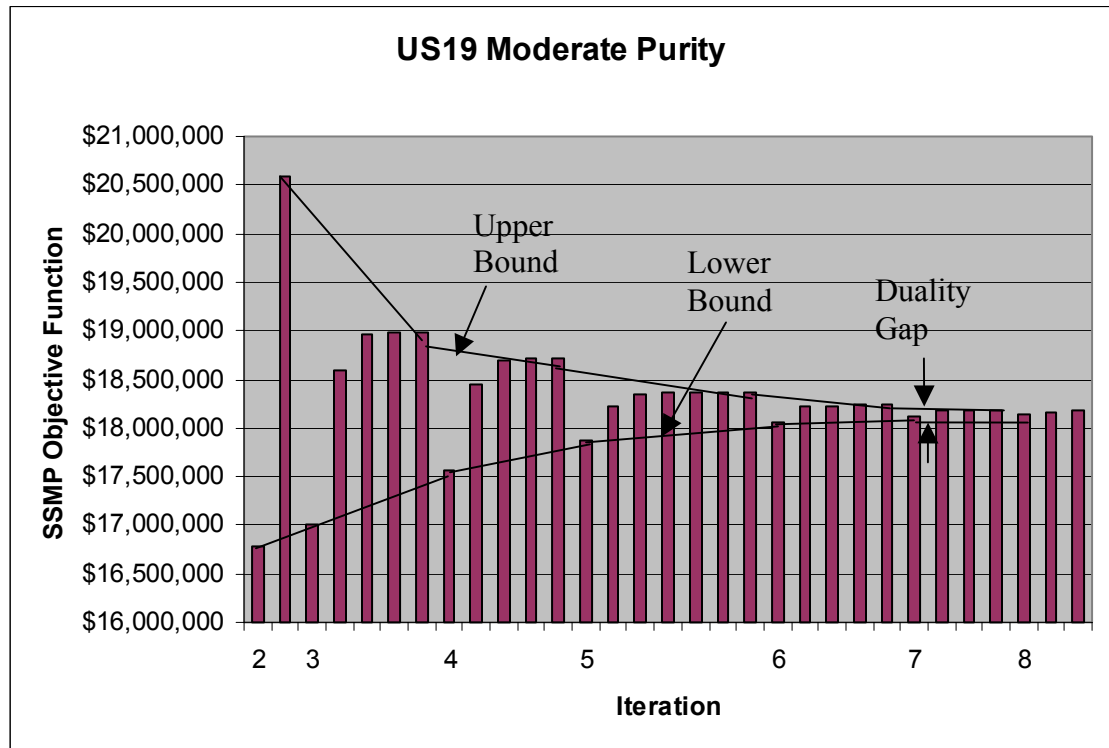


Figure 3.16: SSMP objective function value versus iteration.

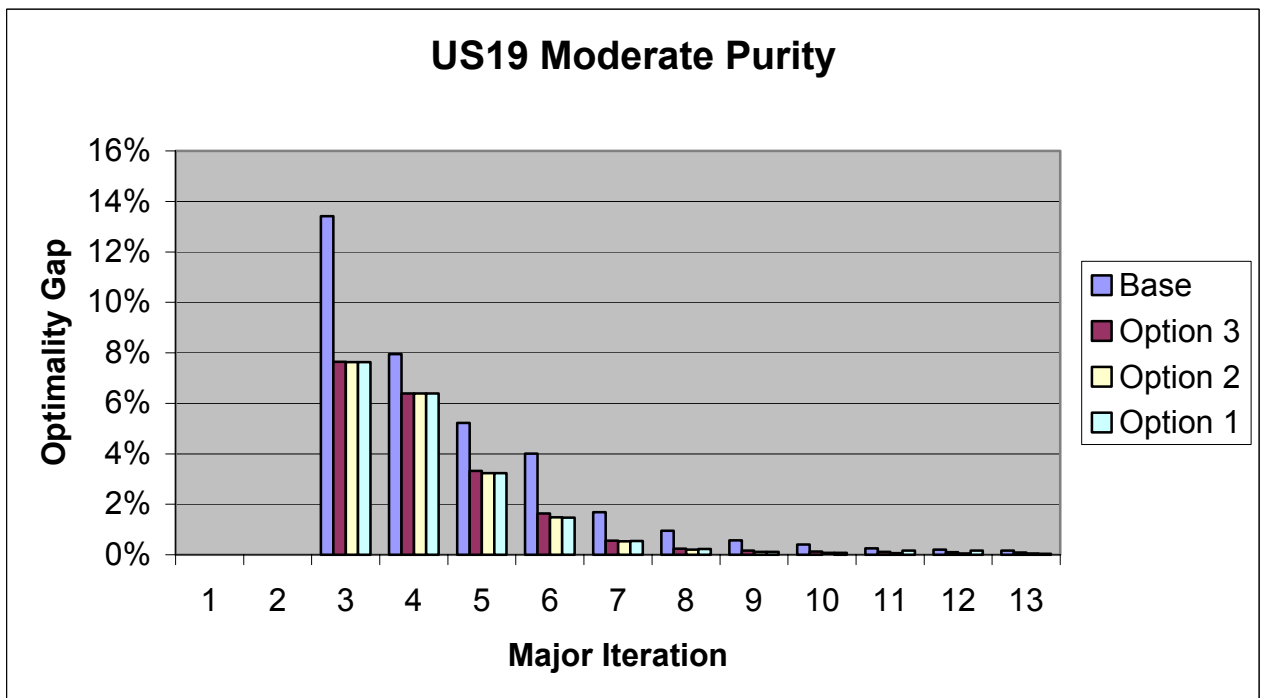


Figure 3.17: Optimality gap vs. iteration using SSMP (US19).

Note that the dual improvement approach provides reliable method of skipping some subproblems. There is an optimal plan associated with any dual solution. If for a given station group, the same plan is optimal for all dual solutions in the SSMP, then this plan will be optimal for every convex combination of these dual solutions. In this case, the plan generation subproblem will not generate a new plan. We can skip this subproblem with no impact on solution quality. We refer to this as the Linearity Check. The Linearity Check reduces the number of minor iterations and column generation time. Its effect is similar to that of skipping subproblems if new columns are not generated in a given iteration. However, with skipping subproblems, there is a chance of missing some new plans, this chance does not exist with the linearity check.

Also note that minor iterations run until no new columns are generated. As a result, there is a high number of minor iterations. We investigated the impact of stopping the minor iterations when the maximum reduced cost associated with the most recently generated columns falls below a specified percentage of the objective function value. As expected, the number of minor iterations is reduced. However, the benefit is offset by an increase in major iterations. Clearly the plans generated in late minor iterations have an impact on solution quality. In the all cases after this point we run the minor iterations to optimality.

### **3.9 Integer Solutions**

In this section, we compare FAM and SDM performance in finding integer solutions. SDM solutions are convex combinations of integer subproblem solutions; this formulation is a tighter relaxation of the original problem than FAM, and we would expect SDM to perform better than FAM in finding integer solutions. Unfortunately there are several factors working against SDM. First, the SDM LP solutions are more fractional than FAM. Second, if we have generated many plans in the LP phase, SDM may contain more integer variables than the original FAM formulation. Third, since we do not generate all possible columns, the SDM LP solution does not guarantee an optimal or necessarily feasible integer solution. We use the following approaches to get good integer solutions:

- Variable Fixing
- Branch on subproblem variables using Special Ordered Sets
- Fix and Price heuristic

We use a simple variable fixing strategy for both FAM and SDM. The lower bound is set to 1 if the LP solution for an assignment variable exceeds 0.99. In addition, in SDM the upper bound is set to 0 if the reduced cost of a plan assignment is very negative. Variable fixing provides some improvements for both FAM and SDM. More aggressive variable fixing (using a cutoff of 0.95 rather than 0.99) tends to result in infeasibility in the MIP.

The SDM efficiency is improved by branching on variables in the subproblems. In the SDM Master, the plane count constraint is typically binding. Each plan consumes aircraft either overnighing at this spoke station or in the air to/from the hubs. We use special ordered sets (SOS1) to partition the plans for each spoke/aircraft type based on the number of aircraft used. The use of SOS1 provides up to a 50% reduction in processing time.

Tables 3.11 and 3.12 summarize the LP and MIP results for FAM and SDM. In the Star7 cases, SDM is comparable to FAM in the Base and Moderate cases. In the Maximum case, SDM is not only faster, it also produces a better solution. This is due to the tighter MIP formulation and the complete enumeration of columns. In Int7, SDM is slightly worse than FAM in the Base and Moderate cases and again better in the Maximum case. In the US cases, SDM is worse in the Base cases, better in the Maximum and Moderate cases. In the US19 Base and Moderate cases, there is a significant MIP gap, causing a fall-off in solution quality. This is due to the fact that plans are generated only in the LP phase of the solution. Some of these plans are not of very high quality to the MIP solution. We address this issue in the next section.

Table 3.11: FAM MIP performance.

Schedule	Purity	FAM Time (secs)		B&B	Profit	
		LP	MIP	Nodes	MIP	Gap
Star7	Base	6.30	2.31	1	65.38	0.00%
	Max	23.80	64.94	9	63.76	0.60%
	Mod	13.52	7.75	1	65.31	0.01%
Int7	Base	12.64	7.24	1	82.54	0.01%
	Max	45.39	0.16	1	81.43	0.00%
	Mod	31.56	23.77	8	82.40	0.03%
US7	Base	11.83	0.52	7	17.52	0.00%
	Max	184.58	4.19	3	14.67	0.01%
	Mod	91.22	1.52	6	16.66	0.01%
US19	Base	253.00	8.53	20	19.36	0.01%
	Max	8306.73	379.50	30	15.81	0.20%
	Mod	2019.75	147.80	32	18.16	0.04%

Table 3.12: SDM MIP performance.

Schedule	Purity	Hubs	SSMP	SDM Time (secs)		B&B	Profit	
			Option	LP	MIP	Nodes	MIP	Gap
Star7	Base	27		6.83	1.64	1	65.38	0.00%
	Max	3		1.28	0.62	1	64.15	0.00%
	Mod	27		15.62	10.75	1	65.31	0.00%
Int7	Base	26	1	42.90	22.44	8	82.50	0.05%
	Max	26		12.21	0.11	1	81.43	0.00%
	Mod	26	3	44.89	29.38	15	82.34	0.10%
US7	Base	31		23.26	3.67	7	17.50	0.00%
	Max	11	2, 3 sols	14.60	6.11	39	14.66	0.11%
	Mod	11	3	17.53	8.09	36	16.65	0.10%
US19	Base	31	2, 3 sols	386.20	235.24	163	19.33	0.18%
	Max	11	2, 3 sols	97.61	77.56	90	15.79	0.29%
	Mod	11	1	340.11	323.36	318	18.06	0.55%



### **3.10 Improving MIP Performance with a Fix and Price Heuristic**

SDM does not produce good feasible integer solutions in all cases. This situation is not uncommon when the LP is solved using column generation and then the MIP is solved without generating more columns. Approaches to improve the quality of integer feasible solutions include tightening the LP formulation and/or generating columns in the MIP processing using branch-and-price (Barnhart et al. 1998).

Another approach that has been used successfully is to sequentially fix variables whose LP values are close to 1 and resolve the LP. If the LP objective value degrades beyond a specified limit, new columns are generated using the current duals (Krishna et al. 1995). This approach works well when we are concerned primarily with solution quality. In the FAM problem, we have global resource constraints (plane count) and need to ensure that the integer solution is feasible relative to them. Our implementation addresses feasibility by allowing some variables to be unfixed. In the context of a branch and price tree, we dive down the tree by fixing candidate variables. If we hit a poor quality solution (either an infeasible solution or one with profit less than a specified threshold) we unfix variables with low reduced cost (These are the plans that are consuming plane count). In the unfixing step we add a cut to avoid cycling. Define  $\Phi$  as the set of plans that have been fixed. We add the  $\sum_{i \in \Phi} p_i \leq |\Phi| - 1$  to ensure that as plans are fixed, we do not consider a solution that has already been identified as poor quality or infeasible. We then continue diving until we reach a feasible solution with a plan fixed for every station group. There may assignment variables for the hub to hub flights that are still fractional. At this point, we pass the problem including bounds and cuts to a MIP solver.

The algorithm is as follows:

0. Solve SDMmp LP relaxation to get a feasible solution (leave artificial variables in the problem for extra planes).
1. Solve SDMsp to generate new columns, solve SDMmp,
  - a. If SDMmp solution is feasible and one plan has been fixed for each station group then go to STEP 4 and solve the MIP.

- b. If SDMmp solution is feasible and some station groups have not been fixed then go to STEP 2, Variable Fixing.
  - c. If SDMmp objective function is less than a specified threshold then go to STEP 3 unfix some variables.
2. Variable Fixing: Check plan solution values. For any plan with value  $> 0.99$ , set lower bound to 1. If no unfixed plans have value  $> 0.99$  choose the plan with the largest value and fix it's lower bound to 1. Go to STEP 1.
3. Unfix variables: For all fixed plans with reduced cost  $< -9,999$  set lower bounds to 0. Add a cut so that this solution does not recur. Go to STEP 2.
4. Solve the MIP. There may be some non-plan variables that are fractional. Solve the MIP with the current columns and lower bounds set.

Table 3.13 summarizes the result of the “Fix and Price Heuristic.” The fix and price heuristic drives plan variables to integer values but does not directly affect the assignment variables of hub to hub flights, in some cases additional nodes are required to get a completely integer solution (US19 Base and Max). In most cases the number of MIP branch and bound nodes is 1, the problems are solved at the root node. Note that this procedure has no impact on Star7, since it required only one node using SDM.

The MIP times are generally reduced with the fix and price procedure. For Int7 Base, the number of nodes is reduced from 8 to 1 and the corresponding MIP time drops from 22.44 to 1.26 seconds. In addition, the solution quality improves from \$82.50 to \$82.52 (millions per week). We see similar improvements in Int7 Moderate, where the MIP time is reduced and the solution quality improved from \$82.34 to \$82.41. In US7, nodes and MIP times drop in each case. In US19, the LP time increases, MIP time decreases and the solution improves to match that of FAM. In the US19 Moderate case, LP and MIP times drop and the solution improves from \$18.06 to \$18.15 (million per day). Overall the fix and price heuristic is very effective on these problems.

Table 3.13: SDM performance using fix and price heuristic.

Schedule	Purity	SDM Time (secs)			Profit- \$mm		Time	
		LP	MIP	Nodes	MIP	Gap	Total	SDM/FAM
Star7	Base	7.59	1.75	1	65.38	0.00%	8.47	98.37%
	Max	1.34	0.66	1	64.41	0.00%	1.90	2.14%
	Mod	16.48	11.39	1	65.31	0.00%	26.37	123.98%
Int7	Base	38.67	1.26	1	82.52	0.00%	39.93	200.86%
	Max	12.21	0.13	1	81.43	0.00%	12.34	27.09%
	Mod	44.73	3.56	1	82.41	0.00%	48.29	87.28%
US7	Base	25.86	0.22	1	17.51	0.00%	26.08	211.17%
	Max	13.23	1.95	1	14.67	0.00%	15.18	8.04%
	Mod	15.97	0.16	1	16.66	0.00%	16.12	17.38%
US19	Base	834.07	24.52	2	19.36	0.01%	858.59	328.29%
	Max	83.03	32.83	15	15.77	0.33%	115.86	1.33%
	Mod	259.22	1.52	1	18.15	0.00%	260.74	12.03%

Figure 3.18 compares the CPU time of SDM versus FAM. In the Base case SDM is comparable to FAM in the Star network. In the general networks, SDM is less efficient, especially in the large schedules. SDM is much more efficient than FAM in the Maximum purity cases. It is generally more efficient in the Moderate cases.

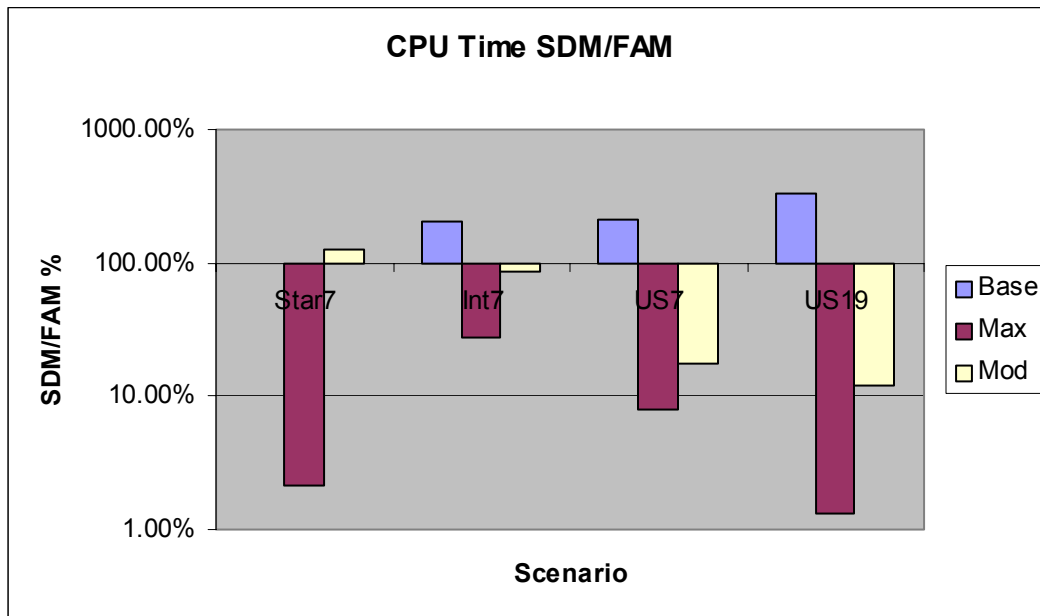


Figure 3.18: SDM vs FAM CPU comparison.

### **3.11 Summary**

In this chapter we investigate the impact of adding station purity constraints to FAM. Station purity can make the FAM LP relaxation more difficult to solve and make the LP solutions more fractional. We develop a station decomposition approach that takes advantage of the hub and spoke network typical of major airlines. The Station Decomposition Model (SDM) isolates the effects of station purity in subproblems. This approach works very well, compared to FAM, on problems in which we can fully enumerate FAM assignments for the spoke stations. This is not possible in problems with large spoke stations and when columns are generated during the solution process. SDM can have relatively slow convergence, an issue common to many column generation applications. As a result, the benefits of station decomposition versus FAM are limited.

We develop a dual improvement approach that provides better dual solutions during the column generation solution process. This can significantly reduce the number of iterations required to achieve good solutions to the fleet assignment problem LP relaxation.

SDM solutions tend to be more fractional than those of FAM; in some cases this results in higher run times and larger MIP gaps. We develop a fix and price heuristic to develop integer solutions more efficiently by sequentially fixing variables and generating additional plans. The time required to produce good solutions for a large US domestic schedule including purity constraints can be significantly reduced using SDM with the dual improvement algorithm and fix and price.

The overall performance of SDM compared to FAM depends on the network structure of the airline schedule and the complexity of station-based constraints. In the Base case, with no purity constraints, SDM performance is comparable to FAM on the star network, it is worse in a general network. In cases with station purity, SDM performance is significantly better than FAM, SDM is up to 20 times faster than FAM.

## CHAPTER 4

### REVENUE MANAGEMENT

#### **4.1 Introduction**

Airlines serve many different passenger types; their value varies widely based on itinerary and fare purchased. Revenue management (RM) is the process used by airlines to determine the number of seats to make available to each passenger type in order to maximize total revenue. Since passenger variable costs are relatively low, maximizing revenue approximates maximizing profit.

Revenue management typically involves three types of controls:

- **Overbooking** determines the total number of reservations that are sold on any flight. Airlines often sell reservations in excess of capacity in order to offset the effects of cancellations and no-shows
- **Seat Allocation** determines the number of seats to make available to full fare and various levels of discount within each cabin
- **Traffic Flow** determines the number of seats to make available to the various itineraries on each flight. In particular this involves the control of seat availability for local versus flow (through and connecting) traffic.

Revenue management can have a significant impact on the number and mix of passengers accommodated on every flight in the network. Typical estimates for the impact of revenue management are to increase total airline revenue by approximately 5% (Smith et al. 1992).

Because RM can have such a significant impact on revenues, the quality of FAM revenue estimates and solutions depends on having a good representation of RM in FAM. In this chapter we'll review basic RM concepts that affect FAM.

## 4.2 Notation

In addition to the notation defined in Chapter 2, we use the following sets, decision variables and data to describe the revenue management process.

### 4.2.1 Sets

$P$ : set of all passenger types, defined by their itinerary and fare class, indexed by  $p$ .

$C$ : set of all fare classes, indexed by  $c$ .

Note that  $p \in i$  will refer to all passenger types on flight leg  $i \in L$  and that  $p \ni i$  refers to all flight legs in the itinerary of passenger type  $p$ .

### 4.2.2 Decision Variables

$alloc_p$ : the number of seats allocated to passenger type  $p \in P$ .

$traf_p$ : the number of passengers carried (traffic) by passengers of type  $p \in P$ .

$\lambda_i$ : the bid price for flight leg  $i \in L$ .

### 4.2.3 Data and Parameters

$Dmd_p$ : the demand for passenger type  $p \in P$ .

$Dmd\_mean_p$ : mean of the demand distribution for passenger type  $p \in P$ .

$Dmd\_std_p$ : standard deviation of the demand distribution for passenger type  $p \in P$ .

$rev_p$ : average revenue per passenger for passenger type  $p \in P$ .

$cap_i$ : the seating capacity of flight leg  $i \in L$ .

### **4.3 Review of RM Literature**

There are several excellent overviews of revenue management in the airline industry, including McGill and van Ryzin (1999), Boyd and Bilegan (2003). Our discussion is limited to the aspects of revenue management that are most relevant to FAM.

#### **4.3.1 Overbooking**

Airlines overbook flights (sell more reservations in excess of physical capacity) in anticipation of booking cancellations and no-shows. While cancellations and no-shows follow predictable patterns, the actual number of passengers who will show up for any given flight departure is random. Overbooking has impact only of high demand flights. If demand is low, there is no opportunity to oversell. Errors in the forecast of cancellation and no-show rates on high demand flights results in either of two undesirable outcomes: 1) too many passengers showing up for boarding results in oversold passengers who must be compensated for their inconvenience; 2) too few passengers showing up means spoiled (empty) seats that could have been sold due to the high demand. The airline overbooking problem has been the subject of statistical and optimization research beginning with Beckman (1958). Airlines typically set overbooking levels to maximize total revenue while maintaining a limit on the total number of oversales. The net impact of the overbooking process is that even on completely sold-out flights, the expected load factor is less than 100%.

Swan (1983) suggested that this process could be modeled very simply in FAM by setting a maximum load factor to a value less than 100%. This approach is in typical use today with the maximum load factor on sold-out flight being 95-97%.

#### **4.3.2 Seat Allocation**

The objective of seat allocation is to maximize the revenue carried by controlling the number of seats made available to each fare class. The basic seat allocation process is typically focused on a single flight leg with multiple passenger types. Passenger types, defined by fare class, have two characteristics of interest for this problem:

- Demand distribution. Actual demand for any flight is discrete. RM models typically assume demand is continuous and follows a normal or gamma distribution. The normal assumption is adequate for distributions with small coefficient of variation, (or k-value):  $cv = \frac{Dmd\_std}{Dmd\_mean}$ . For distributions with small mean and high standard deviation, the normal assumption is invalid due to potential negative demand values. The gamma distribution provides more flexibility without a negative tail. We will typically use the gamma distribution.
- Revenue. We assume that all passengers of the same type, in this case class, have the same value or fare. Revenue can often be used interchangeably with fare. There are some cases in which airlines share part of the fare with other airlines or with distribution partners. We use revenue to represent that part of the fare that the airline keeps.

Airlines typically take the total seat capacity for a flight and then allocate fixed numbers of seats to each class. While an airline can specify the number of seats available for sale for each class, due to the uncertainty in demand, it cannot prescribe the number of seats sold to each class. The number of seats made available to each class sets an upper bound, or truncates, the number of sales by class.

Littlewood (1972) first investigated the problem of determining the allocation for two fare classes to maximize revenue on a single flight leg. He showed that the optimal solution for 2 classes, where  $rev_1 > rev_2$  is to allocate seats such that:

$$p(Dmd_1 > alloc_1) \leq \frac{rev_2}{rev_1} \quad (4.1)$$

The allocations are set so that:  $alloc_1 + alloc_2 = cap$  ;  $alloc_1, alloc_2 \geq 0$  . Sales are allowed in each class until the allocation is reached. Belobaba (1989) generalized this process to multiple classes (passenger types) in a process known as Expected Marginal



Seat Revenue, EMSR. Given  $m$  customer types,  $rev_1 > rev_2 > \dots > rev_m$ , the optimal combined allocation for the  $n$  highest-valued customer types is set such that:

$$p(\sum_{p \in 1 \dots n} Dmd_p > alloc_n) \leq \frac{\overline{rev_{n+1}}}{rev_n} \quad (4.2)$$

where,  $\overline{rev_{n+1}}$  is the demand weighted average revenue of the  $n$  highest-valued customer types.

Vinod and Ratliff (1990) proposed a general formulation for this problem using a piecewise linear approximation of expected traffic as a function of demand and allocation. The Stochastic Seat Allocation Model, SSAM:

(SSAM)

Maximize:

$$\sum_{p \in P} rev_p E(traf_p | Dmd_p, alloc_p) \quad (4.3)$$

Subject to:

$$\sum_{p \in P} alloc_p \leq cap \quad (4.4)$$

$$alloc_p \geq 0, \forall p \in P \quad (4.5)$$

Note that the objective function, Equation 4.3, is the same as the expected traffic function in Equation 2.11.

The optimal solution to this problem satisfies the KKT conditions (Bazaraa et al. 1979):

$$-\frac{\partial E(traf_p | Dmd_p, \overline{alloc_p})}{\partial (alloc_p)} + \bar{\lambda} = 0, \forall p \in P, \bar{\lambda} \geq 0 \quad (4.6)$$

where  $\overline{alloc_p}$  is the optimal allocation for passenger type  $p$  and  $\bar{\lambda}$  is the optimal value of the dual variable on the capacity constraint (Equation 4.5). At the optimal solution we have:

$$\bar{\lambda} = rev_p \frac{\partial E(traf_p | Dmd_p, \overline{alloc_p})}{\partial(\overline{alloc_p})} \quad (4.7)$$

Note that:

$$\frac{\partial E(traf_p | Dmd_p, alloc_p)}{\partial alloc_p} = \lim_{\delta alloc_p \rightarrow 0} \frac{\delta(alloc_p) p(Dmd_p \geq alloc_p)}{\delta(alloc_p)} \quad (4.8)$$

$$\frac{\partial E(traf_p | Dmd_p, alloc_p)}{\partial alloc_p} = p(Dmd_p \geq alloc_p) \quad (4.9)$$

We can use this result to generalize Littlewood's optimality criteria; the optimality criterion from Equation 4.7 becomes:

$$p(Dmd_p \geq \overline{alloc_p}) = \frac{\bar{\lambda}}{rev_p} \quad (4.10)$$

This gives us an optimality criterion for the allocations based on demand, revenue and the bid price  $\lambda$ .

Note that this approach to estimating expected traffic and revenue assumes that the availability of each class depends on the allocation and sales within this class. Suppose we have a nesting structure containing classes 1,...,n, with  $rev_1 > rev_2 > \dots > rev_n$ . Seat availability is found as:

$$SeatsAvailable_i = alloc_i - SeatsSold_i \quad (4.11)$$

Many airlines use a process known as nesting. In a nested structure the availability of seats within classes are linked so that high value classes cannot be sold-out while seats remain available for lower value classes. Figure 4.1 shows a simple example with 100 seats and three fare classes: Full Fare; Moderately Discounted; Deeply Discounted. Note that all 100 seats could be sold to the Full Fare class. A smaller subset of seats (75) is available for sale to the Moderately Discounted class. A still smaller number of seats

(40) are available for sale to the Deeply Discounted classes. The 60 seats not available for sale to the deeply discounted class are protected for sale to the higher value customers.

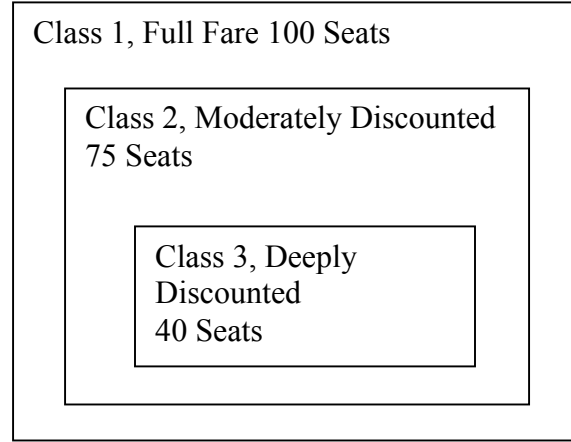


Figure 4.1: Nested class structure.

Suppose we have a nesting structure containing classes 1,...,n, with  $rev_1 > rev_2 > \dots > rev_n$ . The number of seats available in class i, is found as:

$$SeatsAvailable_i = alloc_i - \sum_{j \geq i} SeatsSold_j - \max \left\{ 0, \sum_{j < i} SeatsSold_j - NetAvailability_j \right\} \quad (4.12)$$

$$NetAvailability_j = alloc_{j-1} - alloc_j$$

Williamson (1992) showed that Equation 4.3 overestimates spill when applied to an airline using nested controls. We discuss this approach because this model is the basis for the RM process used by several major airlines. In addition, since we will investigate the incremental revenue changes associated with RM modeling in FAM the assumption of non-nested controls should not materially change our conclusions.

### 4.3.3 The Impact of Seat Allocation on Flight Revenue

We illustrate the impact of revenue management in an example involving a single flight leg with 150 seats, 10 passenger types with mean demand of 15 each and revenue ranging

from \$50 to \$500 per passenger. Table 4.1 illustrates the impact of optimal seat allocations versus no controls in which all passengers are assumed to be of equal (average) value and are accommodated at equal rates. In the Average Revenue case, every class has the same expected traffic; the total expected traffic is 132.19 and total expected revenue is \$37,881. With optimal seat allocations, the space provided to each passenger type depends on the revenue; low revenue results in low allocation and low expected traffic. The total expected traffic drops to 122.57, the total expected revenue increases 4.3% to \$37,881.

Table 4.1: Impact of seat allocations on flight revenue.

Pax Type	Dmd	Average Revenue			Optimal Seat Allocations		
		Rev	Alloc	E(Traffic)	Rev	Alloc	E(Traffic)
1	15.00	\$275	15.00	13.22	\$50	0.00	0.00
2	15.00	\$275	15.00	13.22	\$100	11.08	10.72
3	15.00	\$275	15.00	13.22	\$150	14.15	12.79
4	15.00	\$275	15.00	13.22	\$200	15.67	13.51
5	15.00	\$275	15.00	13.22	\$250	16.69	13.88
6	15.00	\$275	15.00	13.22	\$300	17.46	14.10
7	15.00	\$275	15.00	13.22	\$350	18.07	14.25
8	15.00	\$275	15.00	13.22	\$400	18.57	14.36
9	15.00	\$275	15.00	13.22	\$450	19.01	14.45
10	15.00	\$275	15.00	13.22	\$500	19.38	14.51
Total	150.00	\$36,315	150.00	132.19	\$37,881	150.00	122.57

The impact of RM depends on the level of demand versus capacity. Table 4.2 summarizes total revenue versus capacity. For a given level of demand, the impact of RM is greatest at low capacity. When capacity is 50 seats, the difference in revenue is 48%.

Table 4.2: The Impact of Seat Allocations Depends on the Level of Demand versus Capacity.

Cap	RM Controls		Average Revenue	
	E(Traffic)	E(Rev)	E(Traffic)	E(Rev)
50	48.20	\$20,363	50.00	\$13,750
100	88.87	\$31,842	98.83	\$27,179
150	122.57	\$37,881	132.19	\$36,315
200	144.38	\$40,400	146.00	\$40,151
250	149.14	\$41,111	149.38	\$41,079

Airline demand in any market is a function of price. Table 4.3 illustrates the effects of RM with elastic demand. Demand is high for low fares and decreases as fares increase. The revenues for the high value passenger types are increased so that the total demand and average revenue is the same as the previous case. Again this is illustrated on a flight with capacity of 150 seats. The net impact of seat allocation is greater in this case than in the previous case. The revenue in the RM case increases, average revenue results are unchanged. The net impact of RM is 7.4%.

Table 4.3: Impact of seat allocations with elastic demand.

Path	Dmd	RM Controls			Average Revenue		
		Rev	Alloc	E(Traffic)	Rev	Alloc	E(Traffic)
p1	60.00	\$50	33.95	33.71	\$275	59.99	52.87
p2	26.25	\$100	25.97	23.00	\$275	26.24	23.13
p3	15.00	\$300	19.45	14.52	\$275	15.00	13.22
p4	11.25	\$400	15.32	10.99	\$275	11.25	9.91
p5	9.00	\$500	12.69	8.84	\$275	9.00	7.93
p6	7.50	\$600	10.86	7.39	\$275	7.50	6.61
p7	6.00	\$700	8.88	5.93	\$275	6.00	5.29
p8	5.63	\$800	8.47	5.56	\$275	5.62	4.96
p9	4.88	\$900	7.46	4.83	\$275	4.87	4.30
p10	4.50	\$1,000	6.97	4.46	\$275	4.50	3.96
Total	150.01	\$39,003	150.01	119.24	\$36,315	149.97	132.19

The relative impact when capacity changes is summarized in Table 4.4. At low capacity the impact of RM is to increase revenues by 104%.

Table 4.4: Impact of seat allocations versus capacity with elastic demand.

Cap	RM Controls		Average Revenue	
	E(Traffic)	E(Rev)	E(Traffic)	E(Rev)
50	44.74	\$28,036	50.00	\$13,750
100	79.20	\$36,261	98.83	\$27,179
150	119.23	\$39,003	132.19	\$36,315
200	143.43	\$40,663	146.00	\$40,151
250	148.97	\$41,125	149.38	\$41,079

Since effects of RM are very high at low capacities and small at high capacities, ignoring the effects of RM by assuming that all passengers have average revenue can result in biased solutions. In particular FAM will undervalue small capacity fleets. As a result, FAM solutions, ignoring RM, will favor large capacity fleets.

#### 4.3.4 Flow Controls: Network Revenue Management

Following the deregulation of the US airline industry the dominant scheduling strategy was the development of highly concentrated hub-and-spoke networks. By connecting passengers at the hubs, any flight into or out of a hub could carry passengers in many different O&D markets. Flights in a hub and spoke system typically carry a mix of local passengers and passengers from many smaller markets via 1 stop and 2 stop connections. A single flight may serve several hundred customer types. The amount of local versus flow (connecting and through) traffic is a good measure of network connectivity and opportunity for O&D RM. Given flight capacity and demand and revenue for each passenger type, O&D RM determines best mix of customers to sell across the entire network. As in the single leg case, the best mix depends on the demand and revenue for each path and total capacity available. Due to connecting demand, the traffic and revenue carried on any flight is a function of its capacity as well as the capacities of upline and downline flights (Simpson 1989). Development of revenue management systems to take advantage of this passenger flow can have significant impact on airline revenue and profitability (Smith and Penn 1988).

The seat allocation model, SSAM, can be generalized to incorporate multiple flight legs and multiple passenger types. This is the Origin, Destination Yield Management model that maximizes revenue, ODYMr:

(ODYMr)

Maximize:

$$\sum_{i \in L} \sum_{p \in i} rev_p E(traf_p \mid Dmd_p, alloc_p) \quad (4.13)$$

Subject to:

$$\sum_{p \in i} alloc_p \leq cap_i \forall i \in L \quad (4.14)$$

In this case we maximize total expected revenue across all flight legs by finding allocations for each path subject to sum of allocations on each leg less than capacity for this leg.

The optimal solution:

$$-\frac{\partial E(traf_p \mid Dmd_p, \overline{alloc_p})}{\partial (alloc_p)} + \bar{\lambda}_i = 0, \forall i \in L, p \in i, \bar{\lambda}_i \geq 0 \quad (4.15)$$

or:

$$p(Dmd_p \geq \overline{alloc_p}) = \frac{\bar{\lambda}_i}{rev_p} \forall p \in i, i \in L \quad (4.16)$$

Due to its size and non-linearity, this problem is typically solved by finding bid prices and allocations that are simultaneously primal and dual feasible. The process is summarized below:

1. *Initialize* :  $\lambda_i = r, \delta_i = 0 \forall i \in L, r = \text{a small positive number}$
2.  $alloc_p = \Gamma_p^{-1}(1 - \frac{\lambda_i}{rev_p}) \forall p \in i, i \in L$
3.  $violation_i = \sum_{p \in i} alloc_p - cap_i \forall i \in L$
4. if  $\max_i(violation_i) < \varepsilon_1$  and  $\sum_i violation_i < \varepsilon_2$  **Go to STEP 7**
5.  $\delta_i = f(violation_i) \forall i \in L$
6.  $\lambda_i \leftarrow \lambda_i + \delta_i \forall i \in L$  Go to STEP 2
7. Calculate  $E(traf_p | Dmd_p, alloc_p)$  from Equation 2.11

Figure 4.2: Algorithm to Solve ODYMr.

This process begins with bid prices,  $\lambda$ , set to a small positive value and the change to bid prices,  $\delta$  set to 0. In Step 2 new allocations are calculated for each path using the inverse of the gamma distribution; this ensures that the optimality criteria in Equation 4.16 is met. Violations of the capacity constraints are calculated in Step 3. The convergence criteria are checked in Step 4. The change in bid prices is determined based on the violations in Step 5. Various strategies are employed to reduce the likelihood of stalling and cycling. The bid prices are update in Step 6. In Step 7, the expected traffic for the final allocations is calculated.

The primal solution provides optimal allocations for the O&D RM problem. The allocations are typically not used in practice due to the large number of passenger types and the resulting controls that must be maintained in the reservation system. The typical practice is to use the dual solutions, bid price for each flight leg. The bid price represents the incremental value of a seat on each flight leg. Any seat that is sold will have a positive impact on revenue and profit if its fare is greater than the sum of the bid prices across the legs in its itinerary. Airlines using bid prices make seats are available for any passenger type whose fare is greater than the sum of the bid prices in its itinerary (Smith 1992, Talluri and van Ryzin 1996). Many major airlines now use some form of O&D



RM controls (Vinod 1995, Smith et al. 1998). This approach is typically credited with increasing revenues by 0.5% to 1%.

Two simplifying assumptions are made in ODYMr to ensure that the problem is tractable. First, we assume that demand for different classes and flights are independent. Second, we assume that demand occurs simultaneously. Research has been conducted to address both of these issues.

The independence of demand between various flights and classes assumes that customers know exactly what they want and buy only that. In reality, customers choose flights and fares from a set of product options; the demand for any product depends on its value relative to other offerings. For example, the demand for full fare is very low while discounts are available. The demand for a noon departure may be low until the 9:00 am flight is sold out. The demand for New York to Los Angeles connecting over Dallas depends on the available non-stop fare or for similar itineraries connecting over Chicago. Optimal RM decisions depend on this type of customer behavior. For example, an airline may decide to stop selling discount seats early if it knows that a significant percentage of those customers demanding the discount will buy full fare (upsell) or will buy a ticket on another one of this airline's flights (recapture). Analysis and modeling of this behavior is becoming more practical with the availability of detailed shopping data (Smith 2004). Talluri and van Ryzin (2004) have developed an approach to incorporate customer behavior directly into RM optimization model.

Demand for flights occurs over a booking horizon that may extend up to a year prior to departure. Assuming that demand occurs at a single point in time simplifies the problem for computation but also reduces some of the benefits of RM. Several approaches have been proposed using Markov decision processes that set optimal seat availabilities based on current availability and future demand (Guenther 1998, Subramanian et al. 1999).

While there is great promise associated with the application of customer choice modeling and Markov decision processes there is no evidence that they are implemented at airlines

using FAM. In Chapter 5 we investigate modeling the leading RM processes into FAM. As RM processes and technology evolve, the modeling of RM within FAM will need to evolve as well.

## **CHAPTER 5**

### **INCORPORATING REVENUE MANAGEMENT INTO FAM**

#### ***5.1 Introduction***

The benefits of accurately modeling revenue management in FAM are demonstrated by Smith et al. (1997), Jacobs et al. (1999) and Barnhart et al. (2002). In this chapter, we review several approaches to incorporating revenue management effects into FAM. We investigate in more detail the performance of one approach and develop significant improvements in solution quality and computational efficiency.

The network revenue management process can be incorporated directly into the FAM formulation with modifications in three areas:

- Decision variables: New variables are created for the allocations for each passenger type
- Constraints: Model the non-linear relationship between expected traffic, demand and allocations for each passenger type. Add a constraint so that the sum of allocations on any flight is less than or equal to capacity
- Objective function: Revenue is a function of passenger traffic rather than flight leg capacity.

The resulting model, ODFAM, is:

(ODFAM)

Maximize:

$$\sum_{p \in P} rev_p E(traf_p \mid Dmd_p, alloc_p) - \sum_{f \in F} \sum_{i \in L} C_{f,i} x_{f,i} \quad (5.1)$$

Subject to:

$$\sum_{f \in F} x_{f,i} = 1, \forall i \in L \quad (5.2)$$

$$y_{f,a,t^-} + \sum_{i \in I(f,a,t)} x_{f,i} - y_{f,a,t^+} - \sum_{i \in O(f,a,t)} x_{f,i} = 0, \forall f, a, t \quad (5.3)$$

$$\sum_{a \in A} y_{f,a,t_m} + \sum_{i \in CL(f)} x_{f,i} \leq N_f, \forall f \in F \quad (5.4)$$

$$\sum_{p \in i} alloc_p - \sum_{f \in F} cap_f x_{f,i} \leq 0, \forall i \in L \quad (5.5)$$

$$x_{f,i} \in \{0,1\}, \forall f \in F, \forall i \in L \quad (5.6)$$

$$y_{f,a,t} \geq 0, \forall f, a, t \quad (5.7)$$

$$alloc_p \geq 0, \forall p \in P \quad (5.8)$$

The objective function, Equation 5.1 incorporates the revenue model from the ODYMr model in Equations 4.13 and 4.14. Equations 5.2 through 5.4 are the same as basic FAM. Equation 5.5 ensures that the sum of allocations is less than or equal to capacity. Note that the capacity for each leg now depends on the assignment variables. The expected traffic function can be approximated with piecewise linear segments; this adds significantly to the problem size. For example in the US Domestic case we have 75,000 passenger types. If we approximate each traffic function with 26 linear segments, this increases the number of non-zero elements in the problem from 200,000 to 2.2 million. This problem becomes impractical to solve. In Section 5.2 we review some of the approaches to solve this problem.

## **5.2 Literature Review**

Several approaches to incorporating RM into FAM have been investigated over the past 10 years. These approaches have dealt with the size and non-linearity of ODFAM through various decomposition approaches.

Farkas (1996) demonstrates that RM has a significant impact on traffic volume and mix and by ignoring these effects FAM can sub-optimal solutions. His analysis shows that it is necessary to model effects of both network flow and stochastic demand to improve FAM performance. He concludes that incorporating RM directly into FAM is not practical. He proposes 3 approaches to this problem:

- Column generation. Where each column represents a complete fleeting solution. The master evaluates traffic and revenue, ensures that allocations do not exceed capacity. The columns are generated using a multi-commodity formulation. Although no computational results are published Farkas states that the subproblem is relatively slow to solve (40 minutes) and is impractical for operational use.
- Leg Class revenue management FAM. Since many airlines do not have full network control in their RM systems, Farkas investigates the impact of leg class revenue management control on FAM. He shows that for a typical airline fare structure, the revenue function could be non-concave. This non-concavity makes this formulation unattractive in terms of computational efficiency.
- Decomposing the flight schedule into subnetworks between which there are limited/no leg-interactions. Fleeting solutions for each sub-network are generated, the traffic and revenue for each sub-network is evaluated with a monte carlo simulation. In the FAM formulation, each of the assignments for a subnetwork is represented by one meta-variable. By starting with a feasible leg FAM solution, this approach should always produce improving solutions. No computational results are available.

Kniker (1998) investigates the interactions between RM and FAM. He develops a Passenger Mix Model (PMM) that given a schedule with known flight capacities and a set of passenger demands with known fare, determines optimal traffic and revenue. PMM includes aspects of customer choice modeling and includes recapture (the probability that a customer who is spilled from one flight leg books one another of the same airline). PMM assumes that demand is deterministic and that the airline has complete knowledge and control of which passengers they accept. PMM could be formulated as a multi-commodity flow problem but due to the large number of passenger types and potential paths this approach is impractical. Kniker reduces the problem by using keypaths, the originally desired itinerary for each passenger. Alternate itineraries are necessary only when passengers are spilled from their preferred itinerary. The problem is solved using column generation, with each column representing passengers spilled from one itinerary and recaptured on another. Kniker formulates the stochastic version but does not present results.

Kniker combines PMM and FAM. The integrated problem, IFAM, is solveable but suffers from increased fractionality versus leg FAM. He improves performance through coefficient reduction and additional cuts, but the MIP is still much more difficult to solve than the corresponding leg FAM MIP. Kniker compares performance of various approaches using a monte carlo simulation model. By comparing models that capture the network effects assuming deterministic demand versus stochastic models that ignore network effects, he shows that if flow demand is at least 25% then capturing network effects is more important than capturing stochastic effects. Knicker does not formulate a version of FAM that addresses both stochastic demand and network effects.

Lohatepanont (2001) continues the analysis of IFAM. He investigates the sensitivity of IFAM to several of the simplifying assumptions in its formulation:

- Demand uncertainty. IFAM assumes that demand is fixed and known. The demands used in FAM are forecasts subject to random and systematic errors
- Imperfect control. PMM assumes that airlines have complete control over which passengers are accommodated

- Recapture rate errors. PMM assumes that recapture rate is known.

Through simulation analysis of IFAM and PMM Lohatepanont shows that while relaxing these assumptions, to make the models more realistic, reduces the benefit of IFAM versus leg FAM, IFAM consistently outperforms FAM.

Erdmann et al. (1997) proposes a sequential approach to the itinerary FAM problem. They solve FAM and then the passenger mix problem. Kliewer (2000) proposes an approach that integrates FAM and RM using simulated annealing. Kliewer uses a neighborhood search strategy, starting with an initial feasible solution and looks for improving assignment swaps. He accepts or rejects new solutions based on a simulated annealing strategy. The revenue is evaluated with a deterministic passenger flow model.

Jacobs et al. (1999) proposes a model, which addresses both network effects and the stochastic nature of demand. They use Benders decomposition to integrate the FAM model with ODYMr (see Section 4.3.4). We refer to this approach as ODFAMr. The revenue associated with any FAM solution depends on the capacity assignment for all flight legs. Given an assignment solution, the revenue is estimated in the ODYMr subproblem. The revenue function is approximated in the master problem with a series of Benders cuts, each cut improves the accuracy of the revenue approximation in the master. When a specified accuracy is achieved in the relaxed master problem, the assignment variables are changed to integer and the MIP is solved.

The Jacobs approach is appealing because it addresses both passenger flows in the network and demand uncertainty. It also provides a method of incorporating the passenger mix optimization model used for revenue management directly into FAM. Unfortunately, this approach can suffer from slow convergence and high fractionality.

In Section 5.3 we review the Jacobs formulation in more detail and investigate its performance characteristics. In Section 5.4 we propose extensions to this formulation that improve convergence and reduce fractionality of the solutions.

### 5.3 The ODFAMr Process

As stated in Section 5.1, solving FAM in a network is difficult because the total revenue is a non-linear function of the capacity on every flight leg. From Equations 5.1 and 5.5 we can see that total revenue depends on the allocations for each passenger type; the allocations are constrained by capacity. We can estimate total network revenue for any capacity solution but we don't have a representation of the full revenue surface that can be modeled efficiently in the FAM objective function.

In the Jacobs approach, the ODFAMr revenue function is approximated with a set of hyperplanes. Each hyperplane is determined based on the ODYMr solution for a given set of flight leg capacities in the network. The  $k$ th ODYMr solution consists of total network revenue,  $Rev^k$ , and bid prices for each flight leg,  $\lambda_i^k \forall i \in L$ . The  $k$ th cut is:

$$TNR \leq r_0^k + \sum_{i \in L} \lambda_i^k cap_i \quad (5.9)$$

$$r_0^k = Rev^k - \sum_{i \in L} \lambda_i^k cap_i^k \quad (5.10)$$

where,

$TNR$  is the total network revenue,

$Rev^k$  is the revenue for the  $k$ th ODYMr solution,

$r_0^k$  is the intercept  $r_0^k$  for the  $k$ th cut,

$cap_i^k$  is the capacity for flight  $i \in L$ , in the  $k$ th FAM solution,

Figure 5.1 illustrates this process for a single flight leg with 3 revenue cuts. The actual revenue function for this leg is  $Rev(cap)$ . Three FAM solutions have been found,  $cap^1$ ,  $cap^2$ , and  $cap^3$ . In the FAM master, the revenue function is approximated by the three cuts:  $TNR \leq r_0^k + \lambda_i^k cap_i, k = 1, 2, 3$ .



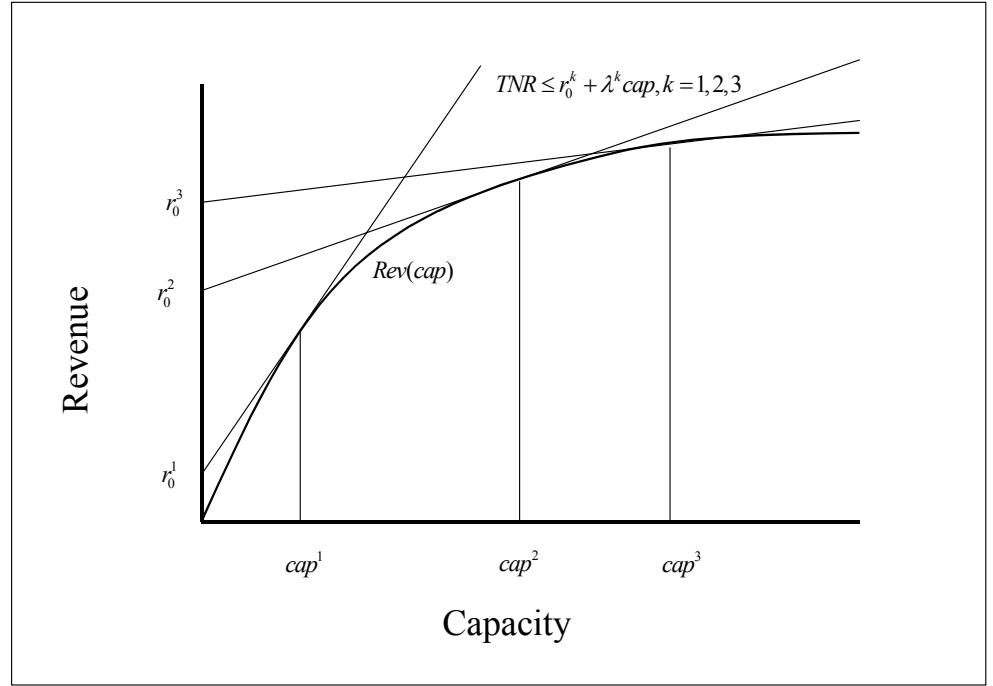


Figure 5.1: Approximating the ODFAMr revenue function on a single flight.

The quality of the revenue approximation is measured by the gap between the approximate total network revenue,  $TNR_{LP}(cap)$ , as defined by the cuts, and the actual revenue from ODYMr,  $Rev(cap)$ . The revenue gap is defined as:

$$RevGap = \frac{TNR_{LP}(cap) - Rev(cap)}{Rev(cap)}.$$

The revenue gap is used as a stopping criteria for the

ODFAMr LP.

The formulation of the ODFAMr master is as follows:

(ODFAMr)

Maximize:

$$TNR - \sum_{f \in F} \sum_{i \in L} C_{f,i} x_{f,i} \quad (5.11)$$

Subject to:

$$\sum_{f \in F} x_{f,i} = 1, \forall i \in L \quad (5.12)$$

$$y_{f,a,t^-} + \sum_{i \in I(f,a,t)} x_{f,i} - y_{f,a,t^+} - \sum_{i \in O(f,a,t)} x_{f,i} = 0, \forall f, a, t \quad (5.13)$$

$$\sum_{a \in A} y_{f,a,t_m} + \sum_{i \in CL(f)} x_{f,i} \leq N_f, \forall f \in F \quad (5.14)$$

$$TNR \leq r_0^k + \sum_{i \in L} \sum_{f \in F} \lambda_i^k cap_f x_{f,i} \quad (5.15)$$

$$x_{f,i} \in \{0,1\}, \forall f \in F, \forall i \in L \quad (5.16)$$

$$y_{f,a,t} \geq 0, \forall f, a, t \quad (5.17)$$

$$TNR \geq 0 \quad (5.18)$$

This formulation is very similar to that of basic FAM. The revenue terms in the objective function, Equation 5.11, are replaced by a single decision variable, total network revenue,  $TNR$ . Total network revenue is constrained by the revenue cuts in Equations 5.15.

On the first iteration, if there are no revenue cuts,  $TNR$  and this model are unbounded.

We can put an upper bound on  $TNR$  to get a feasible solution. An alternative is to solve leg FAM on the first iteration. In the ODFAM context, this is equivalent to adding a leg revenue cut:

$$TNR \leq \sum_{i \in L} \sum_{f \in F} R_{f,i} x_{f,i} \quad (5.19)$$

The solution to this problem is passed to the RM model, an initial cut is produced and added to the FAM master. Equation 5.19 is then dropped from the master. Because the

leg FAM revenue may be lower than the RM revenues the revenue gap can be negative on the first iteration.

Figure 5.2 summarizes the flow of the FAM and RM models. After the initial FAM master solution, the leg cut is dropped, the RM model is solved and the initial revenue cut is added to the master. The ODFAMr master and RM models are solved and cuts added until the revenue gap reaches a satisfactory level. The revenue gap goal is generally specified as a percentage of the RM revenue. Typical stopping criteria range from 0.1% to 1.0% of total revenue.

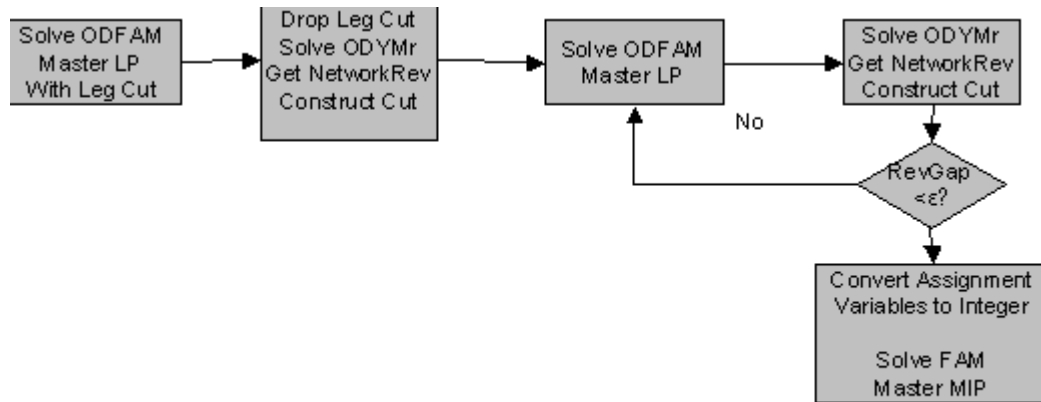


Figure 5.2: ODFAMr model flow.

While benefits from ODFAMr versus leg FAM have been reported, ODFAMr suffers from slow convergence in problems where there is a wide range of possible capacities for individual flight legs. For example, in the US7 scenario there are flights on which capacities can range from 32 to 225 seats. We will show that this range can make the revenue approximation poor. The convergence of ODFAMr on the US7 scenario is illustrated in Figure 5.3. We can see that after 20 major iterations, the revenue gap is approximately 20%. Any gap greater than 1% can cause degradation in solution quality. Figure 5.4 shows the impact of ODFAMr on profit versus the basic FAM solution. Again in this case the errors due to poor revenue approximation result in reduced profit versus basic FAM.

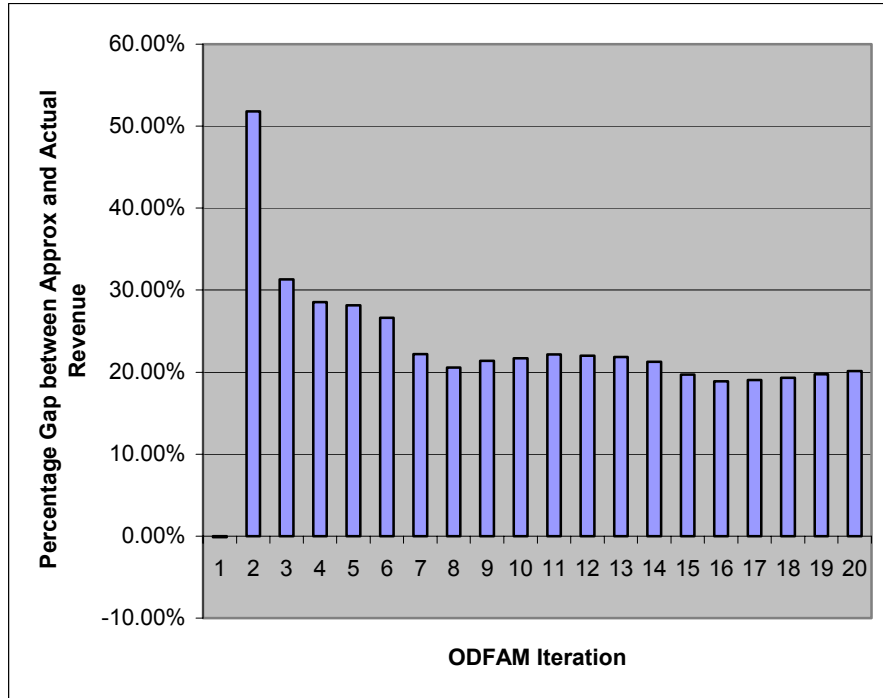


Figure 5.3: ODFAMr revenue gap versus major iteration, US7.

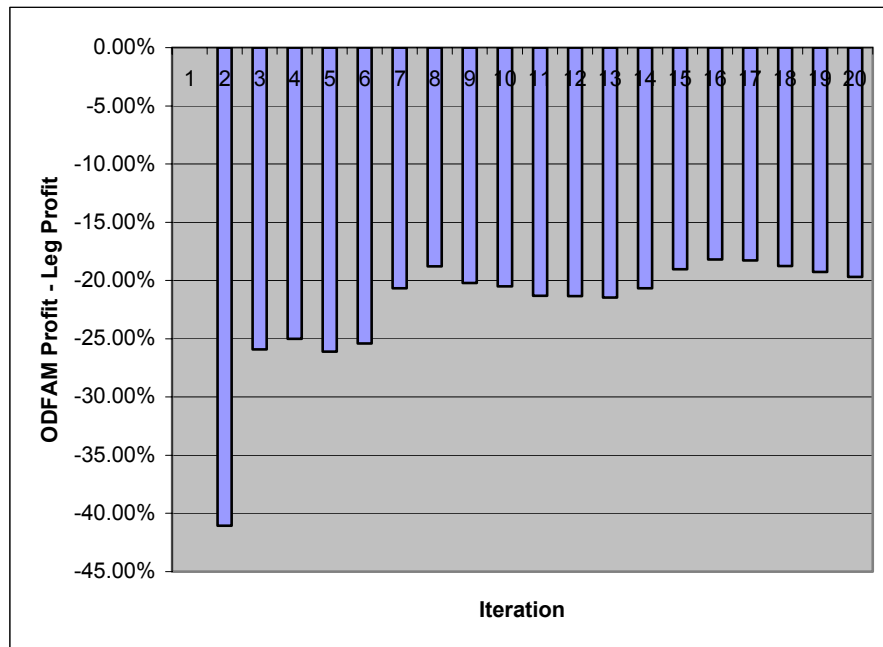


Figure 5.4: ODFAMr profit versus FAM, US7.

The slow convergence is due to the nature of the revenue approximation. The cuts provide an upper bound on revenues; at or near previous solutions, the approximation is good, as we move away from previous solutions the approximation degrades. Since every error is an over-estimate of actual revenue, the ODFAMr master is motivated to move to regions away from previous solutions, where revenue can be significantly overestimated. When the capacity range for any flight is limited, the revenue errors are relatively small but when capacity can vary widely then errors are large. There are two situations that drive large errors:

- Flight legs with small capacity and high bid prices. The model thinks it can increase capacity and that each seat has a high marginal benefit. The actual revenue function is concave, where each additional seat has a smaller marginal revenue benefit. If the model moves to a very high capacity it creates a large revenue gap.
- Flight legs with large capacity and low bid prices. The model thinks it can reduce capacity without reducing revenues. As capacity is reduced the marginal cost of each lost seat increases. If the model moves to a very small capacity, the actual traffic and revenue falls, creating a large revenue gap.

Both of these cases can result in significant revenue overestimates. On any given flight the solutions can oscillate. A small number of flight legs with large revenue gaps can cause the ODFAMr solution to perform poorly. After the LP phase we can have a significant difference in revenue in the ODFAMr master versus ODYMr model. The same is true for the MIP solutions. Table 5.1 summarizes the profit for US7 at various stages of solution for leg FAM and ODFAMr. We see that at the end of the LP phase the ODFAMr master believes that profit is \$19.78 mm versus \$17.51 for leg FAM. The actual revenue from ODYMr is \$13.79 versus \$17.56. Note that the revenue estimate for FAM increases by 0.3%. This increase in revenue is the result of the ODYMr optimization. The revenue and profit errors carry through into the MIP stage.

Table 5.1: FAM and ODFAMr daily profit (\$ millions), US7.

	FAM	ODFAMr
LP Approx Profit	\$17.510	\$19.781
LP Actual Profit	\$17.556	\$13.790
MIP Approx Profit	\$17.510	\$19.758
MIP Actual Profit	\$17.556	\$13.670

Note that in Chapters 5, 6 and 7 profit is measured based on ODYM revenue. In Chapters 1, 2 and 3 Profit is based on revenue in the FAM objective function.

### 5.3.1 Improving ODFAMr LP solutions

The convergence issues shown in the previous section are due to the approximation of the revenue function with a relatively small number of hyperplanes. The dimensionality of the revenue function is on the order of the number of legal flight/fleet combinations minus the number of flight legs. For the US7 scenario this is 27,698. The cuts are sometimes generated in areas that are far from optimal. Neither the ODFAMr master nor the RM subproblem has an accurate estimate of profit. The ODFAMr master has a good estimate of cost but a poor estimate of revenue; the RM model has a good revenue estimate but no cost information. We investigate methods of improving ODFAM performance by expanding the scope of the RM subproblem to include cost. We refer to this model as ODFAMp. The objective of this modified subproblem, ODYMP, is to maximize profit rather than revenue. This results in profit improvements on every iteration and faster convergence.

Capacity is a decision variable in ODYMP. In order to limit the size and complexity of this problem, we restrict it to a convex region bounded by previous FAM solutions. This ensures primal feasibility without bringing cover, balance and plane count constraints into the subproblem. The formulation of ODYMP is:

(ODYMP)

Maximize:

$$\sum_{p \in P} rev_p E(traf_p \mid Dmd_p, alloc_p) - \sum_{k \in K} Fcost^k w_k \quad (5.20)$$

Subject to:

$$\sum_{p \in i} alloc_p - \sum_{k \in K} w_k cap_i^k = 0 \forall i \in L \quad (5.21)$$

$$\sum_{k \in K} w_k = 1 \quad (5.22)$$

$$alloc_p \geq 0 \forall p \in P, w_k \geq 0 \forall k \in K \quad (5.23)$$

where,

$K$  : set of all previous FAM solutions, indexed by  $k$ ,

$Fcost_k$  : FAM cost for solution  $k \in K$ ,

$cap_i^k$  : capacity of flight leg  $i \in L$  in FAM solution  $k \in K$ ,

$w_k$  : weight on solution  $k \in K$ .

The objective function, Equation 5.20, maximizes total network revenue minus weighted costs. We model the traffic function with a piece-wise linear approximation. We use 26 linear segments for each passenger type. Equations 5.21 ensures that allocations do not exceed capacity. Equation 5.22 is the convexity constraint for the weights on the previous FAM solutions.

The ODFAMp master problem is identical to that described in Equations 5.11-5.18; the model flow is identical to that shown in Figure 5.2.

Using the ODYMP subproblem provides significant improvement in the revenue gap and profit performance of ODFAMp. The revenue gap (see Figure 5.5) is reduced to less than 2% after 20 iterations. The ODFAMp profit versus leg FAM increases by 2.6% after 20 iterations, see Figure 5.6. The profit shown in Figure 5.6 is the ODYMP profit. Note that the ODYMP solutions are feasible to the ODFAMp master and are optimal in a convex region that increases in dimensionality on every iteration. As a result, the profit increases on every major iteration.

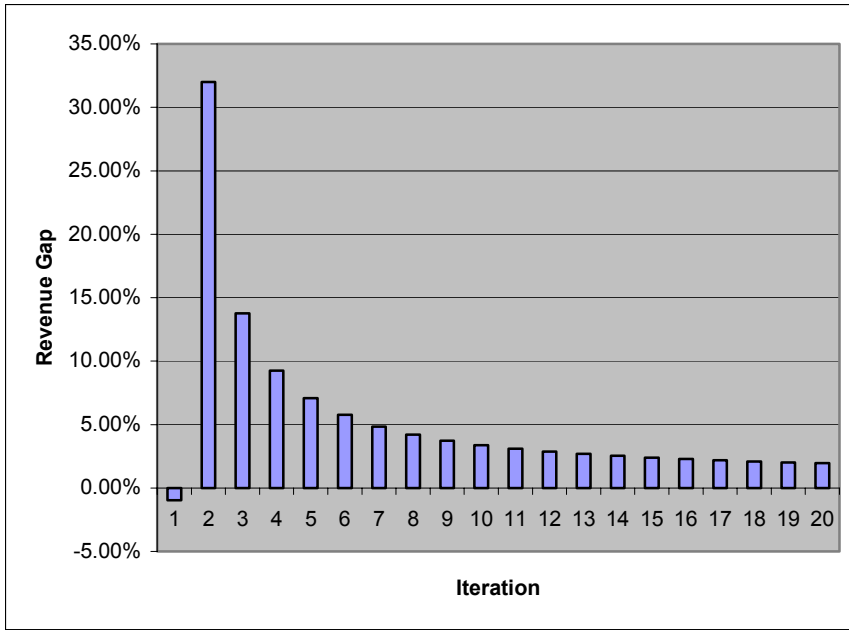


Figure 5.5: Revenue gap for ODFAMp, US7.

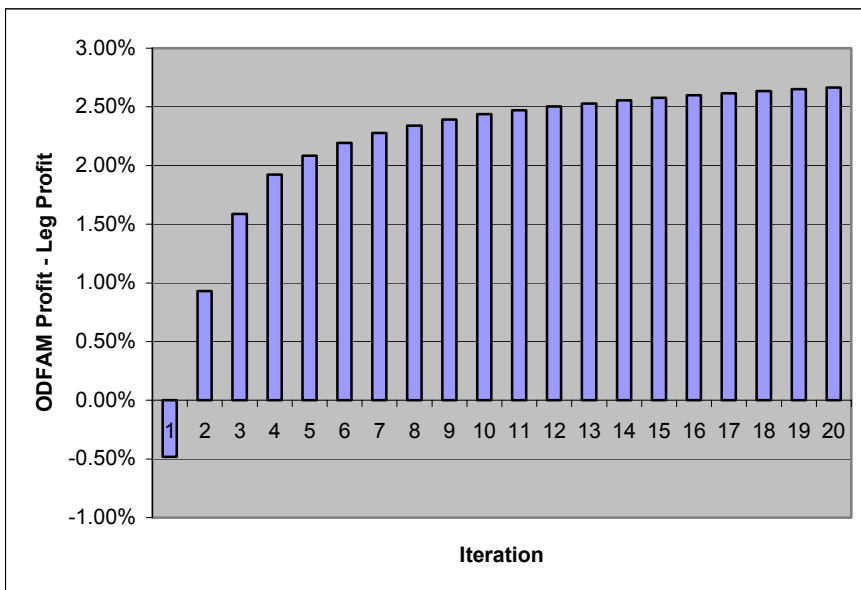


Figure 5.6: ODFAMp profit versus FAM, US7.



While ODYMP provides very good cuts for the ODFAMP master, it is a large model and it is relatively slow to solve. For the US7 scenario, 20 iterations required over 7 hours of CPU time, of this, 5 hours was spent in ODYMP. We can reduce the time spent in ODYMP by taking advantage of patterns in the optimal weights of successive FAM solutions in ODYMP. Table 5.2 summarizes the weights for the first 10 iterations.

Table 5.2: ODFAMP solution weights, US7.

Iteration	Weights									
	Leg	1	2	3	4	5	6	7	8	9
0	1.000									
1	0.865	0.135								
2	0.737	0.084	0.179							
3	0.657	0.060	0.144	0.139						
4	0.624	0.045	0.121	0.107	0.103					
5	0.594	0.034	0.109	0.095	0.085	0.082				
6	0.561	0.026	0.102	0.089	0.078	0.074	0.069			
7	0.539	0.022	0.096	0.083	0.073	0.069	0.062	0.056		
8	0.515	0.018	0.091	0.080	0.068	0.066	0.058	0.051	0.051	
9	0.489	0.016	0.087	0.076	0.066	0.063	0.055	0.048	0.047	0.051

We can make 3 observations:

- The first (leg FAM) solution has the greatest weight; it is close to the optimal solution in the convex region
- The second solution (the first ODFAMP solution) has lowest weight. This is a poor quality solution that results from a very rough approximation of the revenue function
- As other ODFAMP solutions are introduced, they have roughly equal weight.

This pattern is similar in all ODFAMP scenarios that were tested. The optimal solution to ODYMP is in the interior of the convex region bounded by the previous solutions. For every flight in the network, revenue is a concave function of capacity. For any convex combination of capacities, the actual revenue for each flight is greater than or equal to the convex combination of individual solution revenues. Cost is defined only at specific capacities for each flight. For any capacity between these values, cost is a linear function. For any flight, profit, revenue (concave) minus cost (linear) is also a concave

function. Total profit, summing across all flights is concave. So, profit for solutions in the center of this convex region will be greater than or equal to the profit for solutions on the boundary. Using this result, we can implement a simple heuristic to reduce the time spent in ODYMP by choosing solutions in the center of the region. We drop the second solution, give the leg FAM solution weight of 0.5 and give the other solutions equal weight. We refer to this approach as Approximate ODFAMp. The results are shown in Figures 5.7 and 5.8.

Although the revenue gap does not strictly decrease as in the exact case, it reaches a similar level after 20 iterations. The profit improvement of Approximate ODFAMp versus FAM does not strictly increase but it reaches similar levels as the exact case. The ODYMP model is replaced by a heuristic that takes almost no CPU time. As a result, the total time required drops from 7 to just over 2 hours.

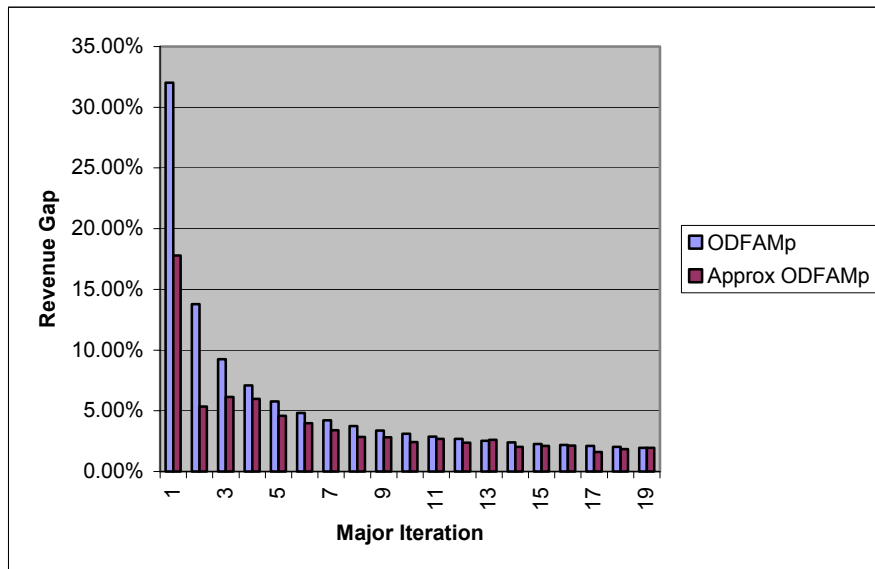


Figure 5.7: Revenue gap for Approximate ODFAMp.

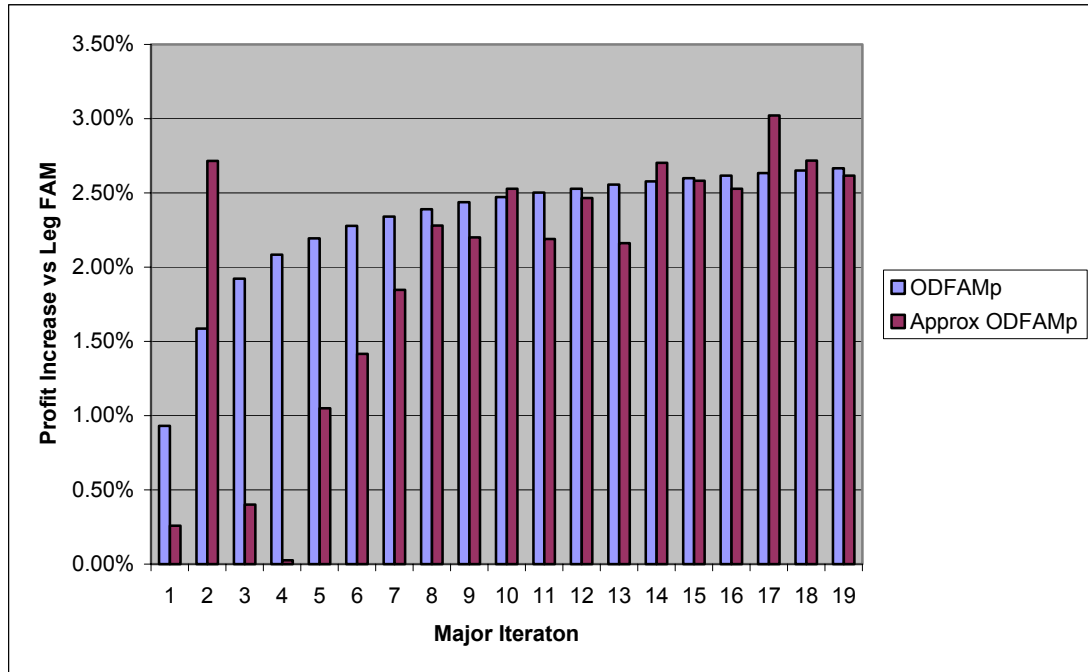


Figure 5.8: Approximate ODFAMp profit versus FAM.

### 5.3.2 Getting MIP Solutions to ODFAM

The ODYMP approach provides significantly improved performance by taking convex combinations of previous solutions. At every iteration, the convex combination solution is better than the last single ODFAMp solution, but if we use the final ODFAMp master model in a MIP then the MIP solution is no better than the last single solution.

While solutions to the FAM problem are often integer or close to integer, the solutions to ODYMP are highly fractional. Even though many of the ODFAMp LP solutions are integer, the convex combinations of different solutions are fractional to the original problem. For US7, the percentage of flights with integer solutions from the relaxed master varies significantly with solution approach:

- Leg FAM 100% of flights have integer solutions in LP
- ODFAM 94%
- ODFAMp 28%.

The solution process for ODFAMp is shown in Figure 5.9. The relaxed master is solved iteratively with ODYMr. After a specified revenue gap is reached, we convert assignment variables to integer and solve the MIP. We integerize the fleets sequentially, starting with the most fractional, the critical fleet. We convert the assignment variables to integer and solve the MIP. In subsequent iterations, the assignment variables for the critical fleets are locked to these values. After solving the MIP, we solve ODYMr and add a new cut to the ODFAMp master. This process is repeated until all fleets are integer.

The results for US7 are summarized in Table 5.3. The quality of the MIP solutions for ODFAMp improves versus ODFAMr but it is still significantly worse than FAM. This is due to the rough approximation of the revenue function in the ODFAMp master problem that is used in the MIP process. Even though after the LP phase, we have a small revenue gap in the ODYMr process, the ODFAMp has a very rough revenue approximation. As a result, master is drawn into regions of poor revenue approximation due to the overestimation of revenues in the branch and bound tree. In the next section we investigate several approaches to improve the MIP solution quality.

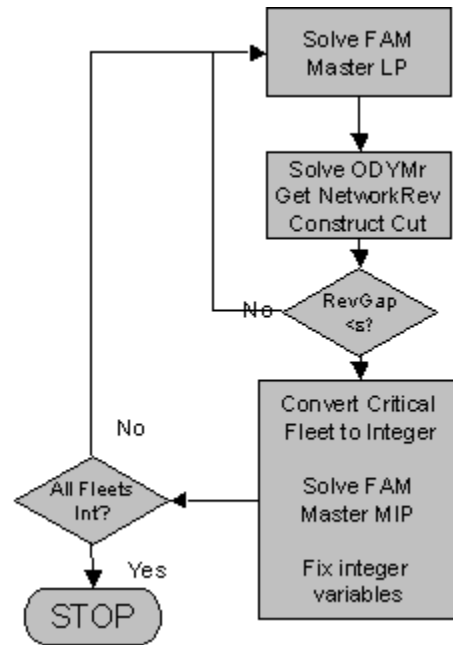


Figure 5.9: ODFAMp model flow.

Table 5.3: Profit performance of FAM, ODFAMr and ODFAMp.

	FAM	ODFAMr	ODFAMp
LP Approx Profit	17.510	19.781	18.374
LP Actual Profit	17.556	13.790	18.023
MIP Approx Profit	17.510	19.758	18.331
MIP Actual Profit	17.556	13.670	15.175

### 5.3.2.1 Penalties

After solving the ODYMP LP for the  $k^{\text{th}}$  time, we have good fractional solutions for each flight leg,  $cap_i^k, i \in L$ . Prior to solving the MIP, we add a penalty function in the ODFAMp master to encourage the MIP to find integer solutions close to the fractional solution:  $penalty = \sum_{f \in F} \sum_{i \in L} \alpha x_{f,i} (|cap_i - cap_i^k|)$ . Where  $\alpha$  is a constant with values \$5 to \$20 per seat. The penalty is subtracted from the profit in the objective function. This tends to make the MIP much more fractional. While this provides improving solutions on small problems (Star7) it is not practically solvable on the large problems (US7, US19).

We can make this process more efficient by making the penalties static. For each flight leg, we add a penalty to the operating costs associated with each potential fleet assignment. The cost is 0 for capacity assignments adjacent to the fractional capacity solution; it increases linearly for capacities above and below the these values:

$$penalty_{f,i} = \begin{cases} 0, cap_{f,i} = cap_i^+ \\ 0, cap_{f,i} = cap_i^- \\ \alpha(cap_f - cap_f^+), cap_{f,i} > cap_i^+ \\ \alpha(cap_f^- - cap_f), cap_{f,i} < cap_i^- \end{cases}$$

where,

$cap_i^+$  = next higher fleet capacity versus optimal LP capacity on flight leg i,

$cap_i^-$  = next lower fleet capacity versus optimal LP capacity on flight leg i.

On small problems the static penalties provide similar profit improvements and greater efficiency than the initial penalty approach; for scenario Int7, the branch and bound tree

contains 15 versus 400 nodes. On large problems, this is solvable but does not provide profit improvements.

### **5.3.2.2 Branch and Cut using Callbacks**

We investigate adding revenue cuts in the branch and bound tree to improve the revenue approximation in the MIP. For any intermediate solution in MIP process, we estimate revenue and add a cut using CPLEX Callbacks (ILOG 2001). We initially investigate adding cuts at the bottom of the branch and bound tree, when integer solutions are found. On small problems, this provides up to 0.1% improvement versus leg FAM. On large problems, we see no improvement versus leg FAM. In fact, there is often no improvement after the first integer solution. We also add additional cuts higher in the tree, when 100 or more variables are fractional. This slows down the process but does not improve solution quality.

Within the branch and bound tree, the revenue cuts are weak and do not provide good bounds on profit. The cuts only affect profit in a very small area around a given solution. Many nodes in the tree are unaffected by previous cuts as a result, we need to evaluate many nodes in the tree. It is not practical to search a significant portion of the tree. Efficiently solving this problem requires a different, stronger cut.

## **5.4 ODFAM<sub>plr</sub>: ODFAM using Prorated Leg Revenue**

In the ODFAM<sub>plr</sub> process, we have good revenue approximations near each previous FAM solution. The quality of the approximation drops when capacity for any flight varies significantly from a previous solution. This is due to the assumption that bid price is constant over the possible range of capacities on each flight. In reality, bid prices change with capacity and this assumption of constant bid price is invalid when the capacity range is large. We make this assumption for two reasons. First, it is easy to construct the cuts. Second, the cuts never underestimate revenues and never cut into the feasible region of the ODFAM<sub>plr</sub> master. We can improve ODFAM<sub>plr</sub> if we can efficiently and accurately estimate revenues for a larger region around a FAM solution.

We propose to use the ODYMr model to estimate the revenue impact of sequentially changing capacity for each flight from the current ODFAMr master solution and then attributing the change in total network revenue to the change on this flight leg.

We use revenue proration to decompose network revenues to a flight leg level. Revenue proration is the process of attributing the revenue associated with a single passenger type to the individual flight legs in its itinerary. Total network revenue is equal to the sum of prorated leg revenues for each flight. For example, 100% of the revenue for a local passenger type is attributed to the one flight leg in its itinerary. For a passenger type with two flight legs, we could: 1) split the revenue equally between the flight legs, this is segment proration; 2) split the revenue based on the mileage of each flight, mileage proration; 3) split the revenue based on the ratio of local fares, straight-rate proration. If we define  $proratedLegRev_{p,i}$  as the revenue for passenger type  $p \in P$  assigned to flight leg  $i \in L$ , then the 3 proration approaches are:

- Segment Proration:  $proratedLegRev_{p,i} = \frac{rev_p}{\sum_{p \ni j} 1}$
- Mileage Proration:  $proratedLegRev_{p,i} = rev_p * \frac{mileage_i}{\sum_{p \ni j} mileage_j}$
- Straight-rate Proration:  $proratedLegRev_{p,i} = rev_p * \frac{\overline{rev_i}}{\sum_{p \ni j} \overline{rev_j}}$

Notation:

$p \ni j$ :  $j \in L$  is a flight leg in the itinerary of passenger type  $p \in P$ ,

$mileage_i$ : flight distance for flight leg  $i \in L$ ,

$\overline{rev_i}$ : demand weighted average local revenue for flight leg  $i \in L$ .

Knicker (1998) states that prorating schemes based on a fixed revenue value per spilled passenger do not perform well in FAM. The Prorated Leg Revenue approach, ODFAMplr, uses the RM model to determine the spill and its value, during the solution process. The use of prorated revenues allows us to attribute network revenue to the

contributions of each flight leg. In the computational tests, we use the straight-rate proration approach.

The iterative solution process for ODFAMplr is similar to that of ODFAM: we solve the ODFAMplr master and then solve ODYMr for this set of capacities to get total revenue and bid prices. Instead of adding a cut assuming that bid price is constant for each flight leg at all capacities, we estimate the bid prices for each flight leg at other possible capacity assignments. On the  $k^{\text{th}}$  major iteration, the current ODFAMplr solution is,  $cap_i^k, \forall i \in L$ . On flight leg,  $i \in L$ , we estimate bid prices for all legal capacity assignments. Let  $\lambda_{f,i}^k$  be the bid price on the  $k^{\text{th}}$  solution for fleet  $f$  assigned to flight  $i$ . When estimating bid prices for flight  $i$ , we hold all other capacities equal to the current ODFAMplr solution. For each bid price, we estimate the allocation for every passenger type flowing over this leg. For each passenger type, the allocation is a function of total bid price across its itinerary, revenue and demand. We assume that the bid prices on the other legs of this itinerary are not changed. The allocation for passenger type  $p$  is found as:

$$alloc_p^k = \Gamma^{-1}(1 - (\lambda_{f,i}^k + \sum_{j \neq i, j \ni p} \lambda_j^k) / rev_p) \quad (5.24)$$

where,

$\lambda_{f,i}^k$ : bid price for the  $k^{\text{th}}$  ODFAMplr solution on flight leg  $i \in L$ , given fleet assignment  $f \in F$ ,

$alloc_p^k$ : allocation for passenger  $p \in P$  on the  $k^{\text{th}}$  ODFAMplr iteration.

Given this allocation we estimate traffic and revenue for this passenger type, and assign revenue to this flight leg.  $rev_{f,i}^k$  is the revenue on flight leg  $i \in L$ , given fleet assignment  $f \in F$  on the  $k^{\text{th}}$  major ODFAMplr iteration:

$$rev_{f,i}^k = \sum_{p \in i} proratedLegRev_{p,i} E(traf_p | Dmd_p, alloc_p^k) \quad (5.25)$$



Given this revenue, we add a cut to the ODFAMplr master:

$$TNR \leq \sum_{i \in L} \sum_{f \in F} rev_{f,i}^k x_{f,i} \quad (5.26)$$

For each flight leg/fleet type combination, we do not resolve the entire  $ODYM_r$  problem.

On the flight leg of focus,  $i$ , we find the minimum bid price so that the sum of passenger allocations is less than or equal to the fleet type capacity on this leg. This is done with a binary search. The allocation for every passenger type is a function of total bid price, Equation 5.25, across each itinerary. We find  $\lambda_{f,i}^k$  such that:

$$\min : \left\{ \lambda_{f,i}^k, s.t. \mid cap_f - \sum_{p \in i} alloc_p^k \mid \leq \varepsilon \right\} \quad (5.27)$$

The procedure to find  $rev_{f,i}^k$  is summarized below:

For each flight leg,  $i \in L$   
 For each fleet  $f \in F$   
   Find min bid price,  $\lambda_{f,i}^k$   
     such that  $\mid cap_f - \sum_{p \in i} alloc_p^k \mid \leq \varepsilon$ ,  $alloc_p^k = \Gamma^{-1}(1 - (\lambda_{f,i}^k + \sum_{j \neq i, j \ni p} \lambda_j^k) / rev_p)$   
   For each passenger type  $p \in l$ , sum leg revenue:  
      $rev_{f,i}^k += proratedLegRev_{p,i} E(traf_p \mid Dmd_p, alloc_p^k)$

Figure 5.10: Algorithm to calculate prorated leg revenues.

The revenue cuts in ODFAMplr, Equation 5.26, are different than those of ODFAM. In ODFAM, we approximate the revenue function with hyperplanes that are tangent to the surface at one point and above at all others. In ODFAMplr, we have good approximations of revenue at every integer solution. For any fractional solution, we approximate the revenue function with a convex combination of the integer revenues. Since the revenue function is concave, we undercut the revenue function for fractional

solutions (see Figure 5.11). While this may cause the LP relaxation to undervalue revenues for some solutions, we do not undercut integer solutions.

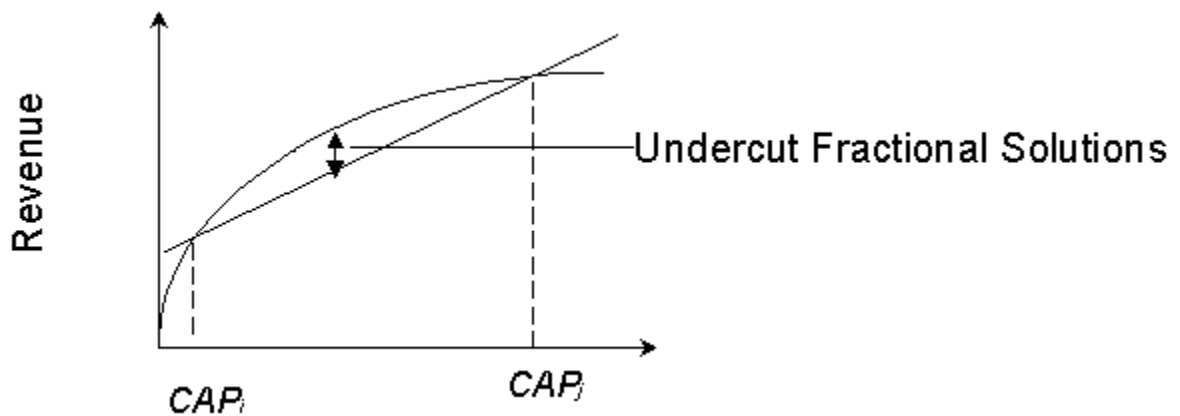


Figure 5.11: Impact of prorated leg revenue cuts.

Note that when we estimate bid prices for various capacities on one flight, we assume that the capacities and bid prices on all other flights do not change. The constant bid price assumption is critical to making this process computationally practical. The alternative is to solve the ODYM problem for every valid flight/capacity combination. In the US7 scenario there are 27,698 valid combinations. The ODYMr process requires on the order of 1 minute to solve and 10 seconds to solve from a warm start. The US7 scenario would require approximately 77 hours to solve all the ODYMr combinations in one major iteration.

Clearly when we change capacity on one flight, the other flights in the network are affected. However, since each flight carries many passenger types with relatively small demand, the effects are distributed and relatively small. In the US7 scenario, there are 240,390 passengers in 172,607 passenger types; 132,607 of these passengers are local. The average demand for flow passenger types is approximately 0.72. Table 5.4 summarizes the impact of changing capacity on 3 types of flights. A capacity change from 32 to 221 seats has a significant impact on the bid price of the affected flight. The traffic and revenue carried by this flight is similarly affected. For example on the hub to

small spoke flight, the bid price changes from \$97.30 to \$0.00 as capacity increases from 32 to 221. Because these effects are spread over many small passenger types their impact is not great on any other single flights. The mean absolute change in bid price on other flights in the network is less than \$0.01. There are 2 flights in the network with bid price change greater than \$1, no flights had a bid price change greater than \$10. The impact is greatest for hub-to-hub flights, where flow traffic is greatest. The mean absolute change in bid price is \$0.05. The maximum bid price impact is \$17.00. While the number of flights with bid price changes is relatively high in the hub-to-hub case, most of these changes are less than \$10; only two flights are affected more than \$10.

Table 5.4: Impact of changing flight capacity on the bid prices of other flights.

Flight Type	Bid Price		Mean Abs Change	Max Change	Number >\$1	Number >\$10
	32 Seats	221 Seats				
Hub to Small Spoke	\$97.30	\$0.00	\$0.00	\$3.87	2	0
Hub to Large Spoke	\$182.11	\$0.17	\$0.03	\$7.38	26	0
Hub to Hub	\$189.86	\$45.53	\$0.05	\$17.00	58	2

While this assumption appears to be reasonable in the US major case it will not be valid for networks with high flow traffic in a relatively small number of passenger types. In this case, changing capacity on one flight will have a greater impact on other flights that share large passenger types. This can be addressed by decomposing the network into isolated sub-networks that do not share passenger flows (see Lohatepanont 2001).

#### 5.4.1 ODFAMplr Formulation

(ODFAMplr)

Maximize:

$$TNR - \sum_{f \in F} \sum_{i \in L} C_{f,i} x_{f,i} \quad (5.28)$$

Subject to:

$$\sum_{f \in F} x_{f,i} = 1, \forall i \in L \quad (5.29)$$

$$y_{f,a,t^-} + \sum_{i \in I(f,a,t)} x_{f,i} - y_{f,a,t^+} - \sum_{i \in O(f,a,t)} x_{f,i} = 0, \forall f, a, t \quad (5.30)$$

$$\sum_{a \in A} y_{f,a,t_m} + \sum_{i \in CL(f)} x_{f,i} \leq N_f, \forall f \in F \quad (5.31)$$

$$TNR \leq \sum_{i \in L} \sum_{f \in F} R_{f,i}^k x_{f,i} \quad (5.32)$$

$$x_{f,i} \in \{0,1\}, \forall f \in F, \forall i \in L \quad (5.33)$$

$$y_{f,a,t} \geq 0, \forall f, a, t \quad (5.34)$$

$$TNR \geq 0 \quad (5.35)$$

### 5.4.2 ODFAMplr Process Flow

The process flow for ODFAMplr is similar to that of ODFAM. After solving master, we solve ODYMr. We add a step to estimate bid prices for other capacities and to add the appropriate cut, see Figure 5.12.

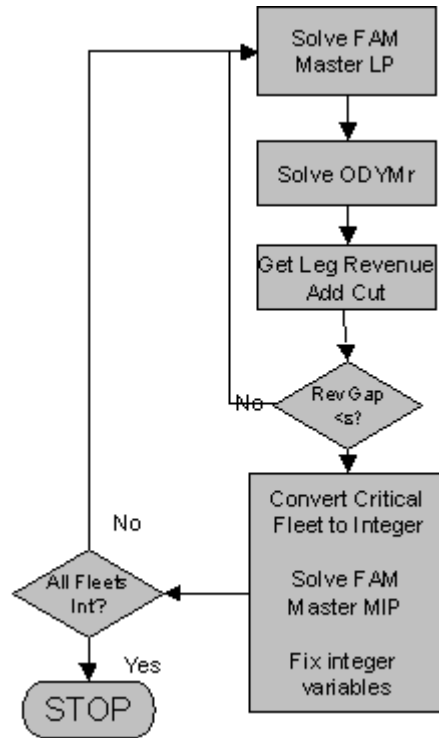


Figure 5.12: ODFAMplr model flow.

### 5.4.3 ODFAMplr Results

We test the ODFAMplr approach initially on the US7 scenario. ODFAMplr has a much better approximation to the revenue function; after 3 iterations the revenue gap is less than 0.01%. The LP profit does not increase as much as the previous ODFAM solutions, since these solutions gained unattainable profit through more fractionality. ODFAMplr profit performance is summarized in Table 5.5. The profit for the ODFAMplr MIP solution does not fall off as with ODFAM. In fact, the ODFAMplr LP slightly underestimates actual revenue and profit. ODFAMplr profit increases 0.3% (0.17 margin points) versus FAM. This corresponds to an annual profit increase of \$20 million.

Table 5.5: ODFAMplr results versus FAM and ODFAM, US7.

	FAM	ODFAMr	ODFAMp	ODFAMplr
LP Approx Profit	17.510	19.781	18.374	17.689
LP Actual Profit	17.556	13.790	18.023	17.579
MIP Approx Profit	17.510	19.758	18.331	17.602
MIP Actual Profit	17.556	13.670	15.175	17.609

The results for INT7 are comparable to those of US7. The revenue gap is less than 0.01% after 3 iterations. The profit increase is lower due low demand (load factor is 49%) and low levels of connecting demand (4%). The resulting profit impact is 0.02% or 0.01 margin point versus FAM.

Table 5.6: ODFAMplr Results versus FAM and ODFAM, Int7.

	Leg FAM	ODFAMr	ODYMp	ODFAMplr
LP Approx Profit	82.535	84.107	84.079	83.931
LP Actual Profit	83.943	83.959	84.061	83.946
MIP Approx Profit	82.533	83.801	84.025	83.931
MIP Actual Profit	83.914	83.784	83.731	83.930

If we increase the level of connecting demand in Int7 by a factor of 5, the profit impact of leg Cuts is 0.05% or 0.04 margin points. In this case, the load factor increases to 60%

and flow demand increases to 18%. This is still not a network with high opportunity to ODFAM approaches. The lack of opportunity is consistent with Kniker's findings that network modeling does not provide significant benefits if flow demand is less than 25%.

The impact of ODFAMplr on US19 is summarized below. Again we see positive, comparable results.

Table 5.7: ODFAMplr Results versus FAM, US19.

	FAM	ODFAMplr
LP Approx Profit	19.353	19.366
LP Actual Profit	19.358	19.364
MIP Approx Profit	19.351	19.343
MIP Actual Profit	19.321	19.342

#### 5.4.4 ODFAMplr Timings

The timings for ODFAMplr versus leg FAM are summarized in Table 5.8. In each case, the LP time for ODFAMplr is higher than Leg FAM due to running multiple major iterations. The number of integer flights after the LP phase is higher for ODFAMplr in each case. Total time includes the LP as well as time required to solve ODYM and find the other bid prices.

Table 5.8: CPU Times Required for ODFAMplr versus FAM, LP relaxation.

LP Solution							
Scenario	Flights	FAM		ODFAMplr			
		Int Flts	CPU Time	Iterations	Int Flts	LP Time	Total Time
Int7	2358	2141	12.64	2	2210	31.75	50.73
US7	4182	4182	11.83	3	4088	101.52	448.83
US19	4182	3851	253.00	3	3858	902.05	1510.41

Performance of the MIP phase is summarized in Table 5.9. In the Int7 case, ODFAMplr MIP performance is slightly better than that of FAM. In the US7 case, the FAM LP solution was integer, ODFAMplr solves relatively quickly but takes longer than FAM.

Table 5.9: CPU Times Required for ODFAMplr versus FAM, MIP.

MIP Solution						
Scenario	FAM			ODFAMplr		
	Time	MIP Gap	B&B Nodes	Time	MIP Gap	B&B Nodes
Int7	7.24	0%	5	10.81	0%	3
US7	0.52	0%	1	5.87	0%	1
US19	8.53	0%	20	816.72	0.01%	4

#### 5.4.5 Summary of ODFAMplr Method

The ODFAMplr approach represents a significant improvement versus ODFAM. ODFAMplr is also more efficient than the ODFAM approaches, the number of iterations required to achieve a low revenue gap is small and the fractionality after the LP phase is low. The amount of time spent in the MIP is comparable to FAM. ODFAMplr consistently provides profit improvements versus FAM.

### 5.5 Summary of Revenue Management and FAM

In this chapter we show that FAM solution quality is sensitive to revenue assumptions and the using average passenger revenues tends to cause FAM to severely under-estimate revenues for flights with high nominal load factor. As a result, FAM tends to undervalue and under-assign small capacity fleets. We demonstrate that the quality of FAM solutions can be improved by incorporating the effects of RM. In networks with significant connecting traffic we can further improve FAM solution quality anticipating the effects of OD revenue management

We demonstrate significant improvements in LP solutions using Benders decomposition. However, ODFAM approaches using Benders decomposition are very fractional and often produce integer solutions that are inferior to Leg FAM. We demonstrate a modification to the Benders approach, ODFAMplr, that produces less fractional solutions. Rather than adding cuts to approximate revenue function we essentially adjust objective function coefficients to reflect RM and network effects across a range of capacities. ODFAMplr out performs Leg FAM in all cases tested. It is also

computationally more efficient than other approaches that tend to increase fractionality of the LP solutions.

American Airlines reports that profit from ODFAMr is typically 0.5 margin points worse than profits from FAM. Tests indicate that ODFAMplr profit increases by approximately 0.3 versus FAM (Jacobs 2004). This represents an increase in annual profit of \$6 million to \$9 million versus FAM.

With ODFAMplr we integrate the RM approach used in production by several major airlines, directly into the fleet assignment process. Since this is not an approximation of RM, but reflects actual RM logic, FAM correctly anticipates the effects of RM on passenger traffic and revenue.



## **CHAPTER 6**

### **FAMILY PURITY, ODFAM and STATION DECOMPOSITION**

#### ***6.1 Introduction***

In this chapter we investigate aspects of purity and ODFAM associated with their application to airline planning. First, in order to reduce its negative impact on profit, we model purity at the crew-compatible family level rather than at the fleet level. This provides many of the benefits of purity in terms of schedule robustness with reduced profit impact. We also investigate the impact of combining purity with ODFAMpl<sub>r</sub>. Finally we investigate the effect of solving this combined problem with station decomposition.

#### ***6.2 Family Purity***

In Chapter 3 we develop an approach for purity by equipment type. The goal is to produce a FAM solution that is robust relative to subsequent planning steps and to operational disruptions. Incorporating purity by fleet has significant impact on both problem difficulty and on solution quality. We show in the US19 cases that moderate purity reduces profit by approximately \$1.5 million per day and maximum purity by up to \$4 million per day. This level of profit impact is unlikely to be acceptable to airlines. Many of the same benefits of fleet level purity can be realized by imposing purity at a family level. Here a family is defined as the fleets that are crew compatible. For example, 737-300 and 737-800 are crew compatible and are contained in the same 737 family. Imposing purity at the family level should yield better FAM solutions while providing benefits in planning, operations, maintenance and capacity swapping.

From the planning perspective, purity at a spoke provides more flexibility in crew scheduling; there are more outbound options for each inbound crew. As a result, the additional costs associated with long layovers and double-overnights can be reduced. Since family purity provides the same flexibility as fleet purity, the benefits for crew planning should be the same.

In operations, purity provides more opportunities for crew move-ups and swaps at the spokes. Since families are defined as crew-compatible, there is no reduction in flexibility by having purity at the family rather than at the fleet level.

In order to support routine and ad-hoc maintenance, airlines must stock spare parts and have appropriately qualified mechanics for each equipment type serving a station. There are many systems, parts and procedures common to fleets within a family. Since family purity reduces the number of families serving the typical spoke station, there is a corresponding reduction in parts and maintenance costs. There is some increase in diversity and cost associated with family versus fleet purity.

Some airlines swap equipment to match capacity to demand. This is typically done within crew-compatible families. Purity at the family level does not reduce flexibility relative to capacity swaps.

The FAM formulation for family purity is similar for that of fleet purity. Variables are added to count the number of families rather than fleets serving a station. The set of Families,  $M$ , is indexed by  $m$ . We add an auxiliary variable  $w_{m,s}$  to indicate whether family  $m \in M$  serves station  $s \in A$  in the FAM solution. We limit the number of families for each station by adding the following new constraints to the basic FAM formulation.

$$w_{m,s} \geq x_{f,i} \forall f \in m, s \in A, i \in L \quad (6.1)$$

$$\sum_{m \in M} w_{m,s} \leq SP_s \forall s \in A \quad (6.2)$$

$$w_{m,s} \in \{0,1\} \forall m \in M, s \in A \quad (6.3)$$

The fleets and families for the Star7, Int7, US7 and US19 scenarios are summarized in Tables 6.1 and 6.2.

Table 6.1: Fleet and families for Star7 and Int7.

Fleet	Capacity	Family
B757	180	B7X
B767	209	B7X
B757	180	B7X
ATR	48	ATR
D9S	97	D9S
M3C	142	MD8
M82	142	MD8
M83	142	MD8
M87	109	MD8
M88	132	MD8
M8D	142	MD8

Table 6.2: Fleets and families for US7 and US19.

Fleet	Capacity	Family
733	128	737
FJT	32	FJT
738	154	737
72S	149	72S
73G	128	737
AT7	66	AT7
757	180	75/76
75P	180	75/76
764	287	75/76
767	204	75/76
76S	252	75/76
M88	142	DC8
M90	148	DC8
CAJ	50	RJ
CJ4	40	RJ
CJ7	70	RJ
CR4	40	RJ
CRJ	50	RJ
RJS	50	RJ

In Chapter 3, we describe the impact of purity on the dispersion of aircraft in the network. This provides a view by equipment type or by family. In this chapter we measure the impact of purity on the FAM solution using two statistics:

- FS – Total number of family/station combinations in solution
- SS – Total number of singletons. The number of family/station combinations with only one arrival and departure for a family.

$$FS_{m,s} = \sum_{m \in M} \sum_{s \in A} w_{m,s} \quad (6.4)$$

$$SS = \sum_{m \in M} \sum_{s \in A} I_{m,s}, \text{ where } I_{m,s} = \begin{cases} 1, & \sum_{f \in m} x_{f,s} = 2 \\ 0, & \text{otherwise} \end{cases} \quad (6.5)$$

Table 6.3 summarizes the impact of family purity on scenarios Star7, Int7 and US19. Each scenario is run with no purity (Base), with moderate purity at both the fleet and family level and then maximum purity at the fleet and family level. The profit in Table 6.3 is measured with ODYMr in order to be consistent with ODFAM. In every case, the negative profit impact of purity is reduced by imposing it at the family rather than at the fleet level. In the Star7 case, moderate purity at the fleet level reduces profit by \$70,000 per week; at the family level there is no profit impact. In the Star7 maximum purity case, the negative profit impact decreases from \$1,600,000 to \$930,000 per week. The benefit of purity, in terms of FS and SS is comparable between fleet and family scenarios. In the Int7 moderate case, family purity reduces the weekly profit impact from \$120,000 to \$0; in the maximum case, the profit impact is reduced from \$1,100,000 to \$70,000 per week. Again, the impact on FS and SS is comparable. In the US19 cases, family purity reduces the negative impact on profit, but it is still greater than \$1,000,000 per day for both moderate and maximum purity.

The LP solution to the family purity problem is generally more fractional than for fleet purity. As a result, the MIP is more difficult to solve. The number of branch and bound nodes required in the US19 moderate case increases from 32 to 201. The family purity maximum case is not solvable. Over 8000 branch and bound nodes are evaluated without finding a feasible integer solution. SDM finds integer solutions for the moderate case in 40 nodes using 1720 CPU seconds; in the maximum family purity case SDM requires 1 node and 1188 seconds to find an integer solution.

Table 6.3: Impact of fleet and family purity on FAM.

Schedule	Purity	Level	FAM			Flt/Sta FS	Singletons SS
			Profit	Nodes	Cplex Time		
Star7	Base		65.79	1	8.61	64	6
	Mod	Fleet	65.72	1	21.27	61	3
	Mod	Family	65.79	1	9.78	62	4
	Max	Fleet	64.18	9	88.74	50	1
	Max	Family	64.86	25	56.22	50	1
Int7	Base		83.91	1	19.88	81	4
	Mod	Fleet	83.79	8	55.33	78	5
	Mod	Family	83.91	1	44.75	79	3
	Max	Fleet	82.81	1	45.55	73	2
	Max	Family	83.84	8	56.08	73	2
US19	Base		19.32	20	261.53	509	124
	Mod	Fleet	18.21	32	2167.55	335	29
	Mod	Family	18.28	201	5216.02	343	32
	Max	Fleet	15.94	30	8686.23	282	30
	Max	Family		>8000	>86000	272	22
	R Mod	Family	19.30	84	621.59	490	111
	R Max	Family	19.16	72	642.08	440	85

Purity, even at the family level continues to have an unacceptable impact on profit in the US19 case. We introduce a more moderate level of purity; strict family purity for stations with 8 or less operations per day, up to 2 families for stations with up to 20 daily operations and no purity for stations with more than 20 daily operations. The results for these relaxed purity cases (R Mod and R Max) are also summarized in Table 6.3. The

profit impact is reduced to \$20,000 per day in the relaxed moderate case and \$160,000 per day in the relaxed maximum case.

### 6.3 ODFAMplr with Family Purity

In this section, we investigate the impact of purity on ODFAMplr. The results are summarized in Table 6.4. In the Star7 and Int7 cases, ODFAMplr provides a slight profit increase versus Leg FAM. In fact, the Int7 moderate case with ODFAMplr generates higher profit than the base case. Maximum family purity is more fractional and more difficult to solve than the fleet based cases. In the Star7 maximum family purity case, the number of branch and bound nodes and CPU time increases by nearly 100 times versus the fleet case. In the US19 relaxed moderate case, profit is increased slightly versus leg FAM; the CPU requirements increase nearly 10 times that of the fleet case. In the relaxed maximum case, ODFAMplr does not increase profit. This is due to the use of critical fleet fixing in the ODFAMplr process. Recall that after the LP is solved, the assignment variables for the most fractional fleet are integerized and the MIP is solved. These assignments are locked in and the next most fractional fleet is solved. Since the family purity solutions tend to be more fractional, the profit loss of locking-in each fleet individually versus solving the entire MIP is magnified.

Table 6.4: Family purity with ODFAMplr.

Schedule	Purity	Level	FAM			ODFAMplr		
			Profit	B&B Nodes	Cplex Time	Profit	B&B Nodes	Cplex Time
Star7	Base		65.79	1	8.61	65.80	1	18.75
	Mod	Family	65.79	1	9.78	65.79	1	12.36
	Max	Family	64.86	25	56.22	64.87	1986	4775.73
Int7	Base		83.91	1	19.88	83.93	3	42.56
	Mod	Family	83.91	1	44.75	83.93	1	45.01
	Max	Family	83.84	8	56.08	83.86	12	65.09
US19	Base		19.32	20	261.53	19.34	1	1718.77
	R Mod	Family	19.30	84	621.59	19.31	1	4870.06
	R Max	Family	19.16	72	642.08	19.13	1	13721.50

## **6.4 Solving ODFAMplr with Station Purity using Station Decomposition**

In this section, we formulate and solve FAM considering purity and OD effects using the station decomposition framework described in Chapter 3. As in the previous section, OD effects are incorporated using the ODFAMplr. Because ODFAMplr requires very few iterations to get a good approximation to the network revenue function, we can incorporate the network revenue effects with limited changes to the SDM process.

Recall that the SDM process described in Chapter 3 has 4 major steps:

- Step 1. Initial plan generation. We generate pure plans and plans based on profit. The profit-based plans are equivalent to assuming that the initial dual values are 0.
- Step 2. Master/Subproblem column generation. We alternatively solve the relaxed master using currently available columns, and then generate new columns using duals from the master. This step gets us good solutions to the LP relaxation of the master problem.
- Step 3. Fix and Price Heuristic. Once we have a good master solution, we fix some variables at integer values and generate new columns. Our goal is to generate columns in regions that are likely to have good integer solutions.
- Step 4. MIP. We convert unfixed assignment variables to integer and solve the MIP.

The ODYM subproblem from used in ODFAMplr, ODYMplr, is applied twice in this combined procedure. First, ODYMplr is run after good LP solutions to the problem have been found; this is between Step 2 and Step 3. So, just prior to starting the Fix and Price heuristic, the revenues for the problem are adjusted to reflect network effects. We apply ODYMplr a second time just before we solve the MIP. This ensures that the integer solution is good relative to network revenues. Although we could apply ODYMplr multiple times in this process, this approach produces good solutions when quality is measured by the final revenue gap. See Figure 6.1.

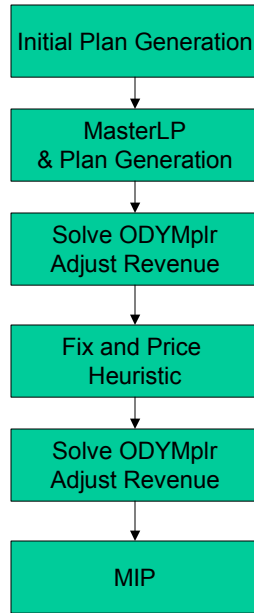


Figure 6.1: SDM\_ODFAMplr -- Station Decomposition with ODYMplr.

Table 6.5 summarizes the comparison between ODFAMplr and SDM\_ODFAMplr. Note that as in Chapter 3, SDM\_ODFAMplr is not run to optimality. A stopping criteria based on the reduced cost of newly generated columns is used. In this comparison Step 3 is stopped when the maximum reduced cost for the columns generated drops below 0.01% of the previous master objective function value. As a result, the profit obtained from SDM\_ODFAMplr is sometimes less than that from FAM.

Also note that in fleet purity cases we can enumerate columns for SDM\_ODFAMplr. With family purity the number of possible plans for even the small pure stations is large. This is due to the potential to mix fleets within a single family and still respect the family purity constraint.

In the Star7 scenario, SDM\_ODFAMplr produces higher profits in all cases. CPU timings are comparable in the base and moderate cases. The Star7 maximum purity MIP is particularly difficult to solve with ODFAMplr. SDM\_ODFAMplr significantly reduces the number of branch and bound nodes required and improves the profit. In the



Int7 cases, SDM\_ODFAMplr runtimes are slightly higher than ODFAMplr; the profit is comparable. SDM\_ODFAMplr performance in the US19 cases is mixed. The base and relaxed moderate cases are solved more quickly, but the solution quality drops. In the relaxed maximum case, SDM\_ODFAMplr solves the problem more quickly and the profit improves. The profit improvement makes back what was lost in the ODFAMplr solution relative to FAM.

Table 6.5: Results for ODFAMplr and SDM\_ODFAMplr.

Schedule	Purity	Level	ODFAMplr			SDM_ODFAMplr		
			Profit	B&B Nodes	Cplex Time	Profit	B&B Nodes	Cplex Time
Star7	Base		65.80	1	18.75	65.81	1	9.40
	Mod	Family	65.79	1	12.36	65.80	1	14.20
	Max	Family	64.87	1986	4775.73	64.92	7	54.69
Int7	Base		83.93	3	42.56	83.92	7	78.07
	Mod	Family	83.93	1	45.01	83.93	1	89.63
	Max	Family	83.86	12	65.09	83.86	1	85.42
US19	Base		19.34	1	1718.77	19.28	1	961.61
	R Mod	Family	19.31	1	4870.06	19.29	8	1675.06
	R Max	Family	19.13	1	13721.50	19.16	4	1465.90

## 6.5 Penalty Costs

The moderate purity constraints that we have used up to this point can limit the mix of fleet and families at small stations. In some cases airlines may want to further limit the dispersion of families throughout their network. In discussions with one major carrier, it has been stated that it could cost up to \$500,000 per year to add a family type to a station. This cost is due to: 1) The need for specialized equipment for each family; 2) The need to stock parts for routine and unscheduled maintenance; 3) The need to maintain mechanics that are qualified to work on each family.

The quality of the FAM solution can be potentially improved relative to these costs by limiting the total number of family station combinations in the solution. We modify the FAM formulation in order to count the number of combinations in the solution and then

add a penalty to the FAM objective function based on this count. This can give us additional control over the dispersion of families at the larger stations. The results using leg FAM are shown in Table 6.6. Base corresponds to no purity, \$0 corresponds to moderate family purity with no penalty. We label the other cases based on the penalty per family station combination per week (Star7 and Int7) or per day (US19). For each case, Table 6.6 contains profit (excluding penalty costs), family station combinations and singletons. The last column in Table 6.6 indicates the reduction in profit per flight station combination reduction.

In Star7, as the penalty is increased to \$30,000 per combination, profit drops from \$40,000 per week; the number of family station combinations is reduced from 64 to 56. The annual profit impact of this reduction is approximately \$2.1mm or \$261,000 per family station combination. Increasing the penalty above \$30,000 further reduces profit and FS. The reduction in flight combinations becomes increasingly difficult and expensive. Reducing FS to 51 costs over \$3mm per combination per year. The results are similar for the Int7 cases, although the cost to reduce family combinations is greater. In the US19 case, there is a cost of \$20,000 per day to impose family purity; this reduces the number of family station combinations from 508 to 488. Imposing a \$1500 per fleet station day penalty reduces the combinations from 508 to 370. The annual cost is \$211,594 per combination. Further reducing family station combinations drives the cost up to \$290,942.

Table 6.6: Impact of family/station combination penalties.

Schedule	Penalty	Profit (mm)	FS	SS	Cost/Combo-Yr
Star7	Base	\$65.79	64	6	
	\$0	\$65.79	62	4	\$0
	\$30,000	\$65.75	56	2	\$261,000
	\$50,000	\$65.59	54	2	\$1,044,000
	\$500,000	\$65.00	51	1	\$3,172,154
Int7	Base	\$83.91	79	3	
	\$0	\$83.91	79	3	\$0
	\$30,000	\$83.86	72	2	\$372,857
	\$50,000	\$83.82	71	2	\$587,250
	\$500,000	\$83.56	70	2	\$2,030,000
US19	Base	\$19.32	508	123	
	\$0	\$19.30	488	104	\$365,000
	\$1,500	\$19.24	370	39	\$211,594
	\$3,000	\$19.21	370	34	\$290,942

While purity can reduce operating profit, the results for the Star7 \$30,000 case and US19 \$1,500 case are probably in the range of interest to airlines based on the statements described earlier. In addition, the reduction in singletons in the US19 case from 123 to 39 makes this solution increasingly attractive for crew scheduling and robustness.

If we assume that each family combination costs an airline \$500,000 per year, then the benefits of adding purity controls can be estimated in terms of the reduction in FS versus profit impact. In the Star7 case, the net impact of reducing FS from 64 to 56 is approximately \$2mm per year. In the Int7 case, the net impact is less than \$900,000. In the US19 case, the annual impact of reducing FS from 508 to 370 is \$24mm.

## 6.6 Summary

In this chapter we introduce the notion of family purity. We modify the FAM formulation to restrict the number of families serving small stations. This problem is generally more difficult to solve than the fleet purity FAM. Family purity reduces the

negative impact on profit compared to fleet purity. The profit levels for family purity are acceptable for Star7 and Int7, however the impact on US19 profit is still too large. We relax the purity parameters for US19 to reduce the profit impact while limiting the number of families serving small stations.

We solve ODFAMplr with purity and show similar benefits to the cases without purity. The purity cases are typically more fractional and some are much more difficult to solve than the base ODFAMplr. We develop an approach to solve ODFAM with purity using station decomposition, SDM\_ODFAMplr. For small cases, station decomposition produces results that are comparable or better than ODFAMplr with purity. This approach reduces runtimes for the most fractional problems. On the large US domestic cases, the SDM\_\_ODFAMplr combination is more efficient; in some cases the solution quality is slightly degraded, in some the solution quality improved.

Finally, we look at the impact of further limiting family/station combinations. In the Star7 and US19, the reduction in family combinations and singletons comes at a cost that should represent a reasonable tradeoff to airlines in terms of profit versus maintenance and crew costs and robustness.

Table 6.7 summarizes the net benefits of ODFAMplr and moderate family purity for three schedule scenarios. In the Star7 case, there is no measurable benefit of ODFAM when purity is imposed; there is a \$2,000,000 benefit due to the reduction in family station combinations. In the Int7 case, the total annual benefit is approximately \$1.9mm. In the US19 case, the total annual benefit is approximately \$27.6mm.

Table 6.7: Total Annual benefits of ODFAMplr and purity.

Schedule	Benefits – Annual Increase in Profit \$mm		
	ODFAM	Family Combination Reduction	Total
Star7	0	2	2
Int7	1	.9	1.9
US19	3.6	24.0	27.6

## **CHAPTER 7**

### **FUTURE RESEARCH**

In this study, we demonstrate that FAM solutions can incorporate station purity and revenue management effects in order to facilitate subsequent planning processes and operations. While this makes the FAM process less efficient, we improve the overall performance of the airline relative to crew costs, maintenance costs, revenue generation and operational dependability.

These results suggest three potential areas of future research. First, we discuss a method to tighten the Benders cuts used to approximate the revenue function in ODFAM. In Chapter 5 we demonstrate that the ODFAMp method can produce good LP solutions by approximating a non-linear objective function with Benders cuts. Unfortunately the cuts can be loose at integer solutions. We may be able to improve MIP efficiency and solution quality by tightening these cuts. A second potential area of research is to use a Dantzig-Wolfe decomposition approach to the ODFAM problem rather than the Benders approach. We review this approach in Section 7.2. Finally, airline planning and operations can be improved by better integration of planning and operational activities. This requires expanding the scope and combining processes that are currently performed separately. In Section 7.3 we review a general framework for planning integration and suggest areas of future development.

#### ***7.1 Improving ODFAM Performance by Tightening Benders Cuts***

This dissertation highlights one of the difficulties in using a Benders decomposition approach to solving a mixed integer problem. We approximate a non-linear objective function with cuts based on the revenue function evaluated at a FAM solution. Since

these are solutions to the LP relaxation of the FAM problem, the cuts are not necessarily tight at integer solutions. As a result, MIP solution quality may be poor.

It may be possible to improve the value of the cuts by adjusting them to be tight for at least one integer solution. Figure 7.1 illustrates this approach on a single flight. The  $k$ th cut is based on the revenue at  $cap^k$ . This cut,  $Rev \leq r_0^k + \lambda^k cap$  is tangent to the revenue function  $Rev(cap)$  and has an intercept term of  $r_0^k$ . This cut is relatively loose at the integer solution capacity,  $cap_n$ ; the  $n$ th capacity option for this flight. We can tighten this cut by reducing the intercept to  $r_0^{k'}$  so that  $Rev(cap_n) = r_0^{k'} + \lambda^k cap_n$ . In this case, the revenue function is undercut at fractional values.

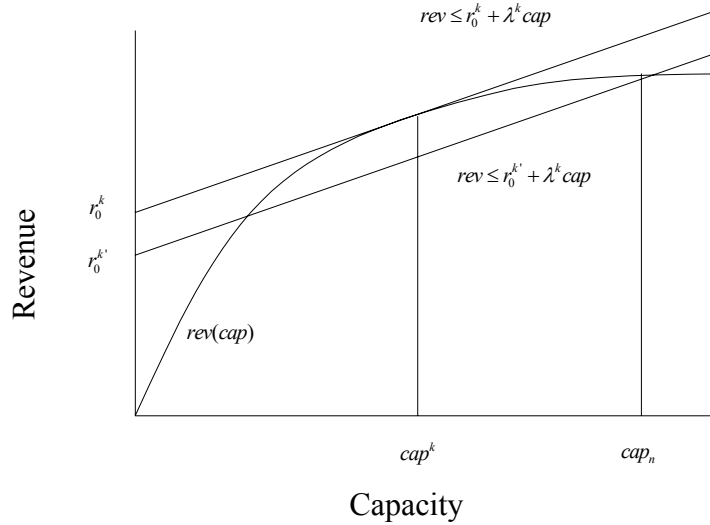


Figure 7.1: Adjusting a revenue cut to be tight at an integer solution.

This process can be extended to a network. Let  $FAM$  be a set of feasible integer solutions to FAM and let  $Fcap \in FAM$  be an integer feasible solution to FAM;

$Rev(Fcap)$  is the ODYMr revenue at this solution. We can find the appropriate value of  $r_0^{k'}$  by solving:

$$\begin{aligned}
& \text{minimize } r_0^{k'} + \sum_i \lambda_i^k cap_i - Rev(Fcap) \\
& \text{subject to } Fcap \in FAM \\
& r_0^{k'} + \sum_i \lambda_i^k cap_i - Rev(Fcap) \geq 0
\end{aligned} \tag{7.1}$$

Equation 7.1 reduces the intercept,  $r_0^{k'}$ , until the cut hits an integer solution. This is equivalent to:

$$\begin{aligned}
& \text{maximize } z = Rev(Fcap) - r_0^{k'} - \sum_i \lambda_i^k cap_i \\
& \text{subject to } Fcap \in FAM \\
& z \geq 0
\end{aligned} \tag{7.2}$$

The revenue function  $Rev(Fcap)$  is evaluated by solving the ODYMr problem described by Equations 4.11 and 4.12:

$$Rev(Fcap) = \left\{ \max \sum_{p \in P} rev_p(alloc_p) \mid \sum_p alloc_p - cap_{i \ni p} = 0, alloc_p \geq 0 \right\} \tag{7.3}$$

Combining 7.2 and 7.3 yields:

$$\begin{aligned}
& \text{maximize } z = \sum_{p \in P} rev_p(alloc_p) - r_0^{k'} - \sum_i \lambda_i^k cap_i \\
& \text{subject to } Fcap \in FAM \\
& \sum_{p \in i} alloc_p - cap_i = 0 \forall i \in L \\
& alloc_p \geq 0, z \geq 0
\end{aligned} \tag{7.4}$$

This removes the capacity terms from the objective function of 7.5 by subtracting a convex combination of the allocation constraints,  $\sum_{p \in i} alloc_p - cap_i = 0 \forall i \in L$ , weighted by the bid prices,  $\lambda_i^k$ .



$$\begin{aligned}
& \text{maximize } z = \sum_{p \in P} (rev(alloc_p) - alloc_p \lambda_{i \ni p}^k) - r_0^k \\
& \text{subject to } Fcap \in FAM \\
& \sum_{p \in i} alloc_p - cap_i = 0 \forall i \in L \\
& alloc_p \geq 0, z \geq 0
\end{aligned} \tag{7.5}$$

Equation 7.7 provides a method to adjust every revenue cut given a set of integer feasible FAM solutions. The adjusted cuts replace each of the original cuts in the FAM master problem. While this does not guarantee integer solutions, it does ensure that the revenue cuts are tight at one or more integer solutions. This improved revenue approximation should improve the performance of ODFAM.

The challenge is to develop a method of efficiently generating a sufficient set of integer feasible solutions to FAM. Techniques to generate these solutions include very large neighborhood search (Ahuja et al. 2001) or constraint programming (ILOG 2001). Solving Equation 7.7 is equivalent to solving the ODYM problem for a given  $Fcap$  solution. It is necessary to efficiently choose the FAM solutions to which Equations 7.7 are applied.

## 7.2 Dantzig-Wolfe Approach to FAM

In Section 5.3.1 we describe a method to generate solutions to the ODFAM LP relaxation. We expand the ODYMr model to maximize revenue in a region bounded by previously generated feasible FAM solutions and we refer to this model as ODYMp. Dual solution values from ODYMp were used to add Benders cuts to the FAM model. ODYMp solutions increase in profit on every major iteration. Finding good MIP solutions using the FAM model is difficult because the approximation to the revenue function is fairly rough even after many major iterations. We propose to improve this process by reformulating it in a Dantzig-Wolfe framework.

Figure 7.2 shows the structure of ODFAM as described in Equations 5.1-5.8. FAM variables include  $x$  (fleet assignment) and  $y$  (ground arc flow); there are three sets of

constraints, balance, plane count and cover. The ODYMr problem consists of allocation variables,  $(alloc)$ , for each passenger itinerary origin, destination fare class, ODF. There is a non-linear function,  $rev_p(alloc_p)$  relating allocation to expected revenue for each ODF. For each leg there is a constraint such that the sum of the allocations on each leg do not exceed capacity. The ODF matrix value is defined as:  $ODF_{p,i} = \begin{cases} 1, & \text{if } p \in i \\ 0, & \text{otherwise} \end{cases}$

The capacity for each flight links the FAM and ODYMr problems together. Capacity is determined by the FAM assignment variables,  $x$ , capacity is used in ODYMr to limit the sum of the allocations on each flight. The ODFAM process uses a Benders framework with the master problem based on FAM and the sub-problem based on ODYM.

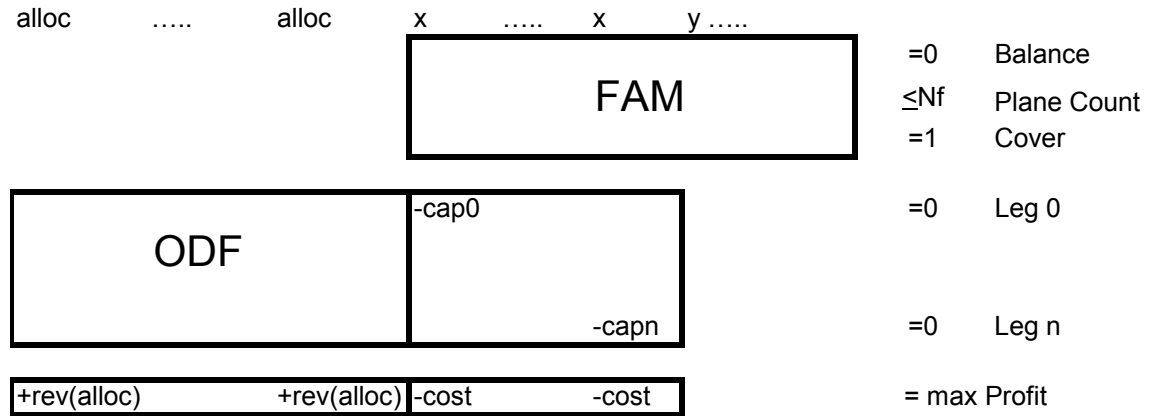


Figure 7.2: ODFAM model, FAM and ODYM as a single model.

In the proposed Dantzig-Wolfe framework, ODYMr is the master; it maximizes total profit in a convex region bounded by FAM solutions. FAM generates columns for ODYMr based on ODYMr dual values. This is similar to the approach we describe in Section 5.3.1, however, rather than adding Benders cuts to FAM on each iteration, we adjust the FAM objective function coefficients based on ODYMr dual solution values. These dual values are the bid prices for each flight leg,  $\lambda$ . Figure 7.3 illustrates the structure of these two models.

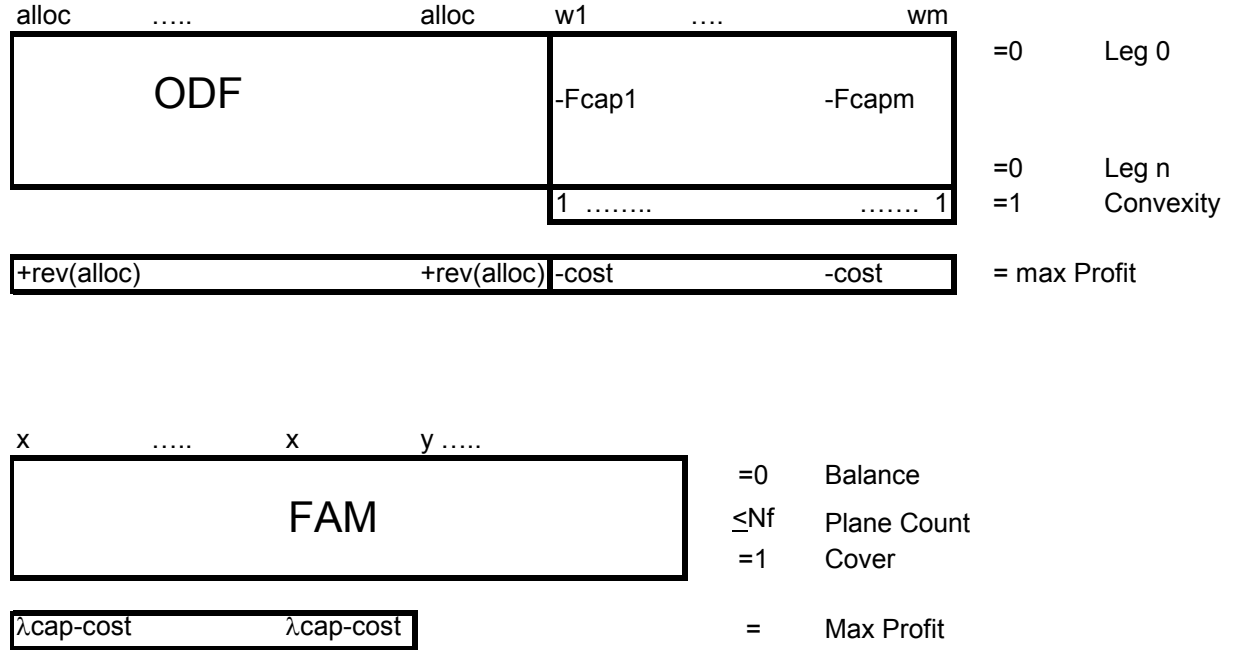


Figure 7.3: Dantzig-Wolfe Decomposition of ODFAM.

The goal of this approach is to improve the quality of the ODFAM MIP solutions. In this approach, the FAM sub-problems are solved to integer solutions. This provides columns for ODYMP that are integer. We also need the ODYMP solution, which is a convex combination of the capacity columns, to be integer. We propose to find integer solutions for ODYMP by applying a branch and price process to both ODYMP and FAM.

We use a branch and cut tree to organize the search process. Each node in the branch and cut tree defines a partition of Fcap columns in ODYMP. Define the following:

$N$  = set of candidate nodes in the tree

$P^*$  = the profit for the current best integer solution

$P_n$  = profit for node  $n$

$F_n$  = set of flight legs with fractional solution values at node  $n$ .

Start by solving leg FAM and adding this solution to ODYMP and to the candidate node set. The solution process is described below:

1. If  $|N| = 0$  STOP. If  $|N| > 0$  pick an unbounded node,  $n$ .
2. Solve ODYMP node,  $n$ . If  $P_n < P^*$  Return to Step 1.
3. If  $|F_n| = 0$  and  $P_n > P^*$ , then  $P^* = P_n$ . This is the new incumbent. Return to Step 1.
4. Pick a leg,  $l \in F_n$ . Create 2 child nodes:
  - a. Left: Add cut to ODYMP and FAM  $cap_l \leq cap_l^-$ , where  $cap_l^-$  is the integer capacity that is next lower than  $cap_l$ . Add this node to  $N$ .
  - b. Right: Add cut to ODYMP and FAM  $cap_l \geq cap_l^+$ , where  $cap_l^+$  is the integer capacity that is next higher than  $cap_l$ . Add this node to  $N$ .
5. Update FAM objective function with current bid prices. Solve FAM for the current node. Add  $Fcap$  to ODYMP. Return to Step 1.

There are three issues to consider in this process. First, there is a set of cuts associated with each node in this tree. The cuts must be managed and applied efficiently. Second, the efficiency of this process will depend on how the tree is explored. The process to pick the next node is critical (Vanderbeck 2000). Third, we must be able to solve ODYMP efficiently.

We propose to solve ODYMP with a primal-dual approach (Johnson and Barnes 1998). We relax the capacity constraints and then adjust bid prices and weights on the capacity solutions in order to achieve primal and dual feasibility. At any point in this process, we have a set of bid prices and associated allocations. These allocations violate the capacity constraints in ODYMP. We solve a Phase 1 problem to find weights on the capacity columns that minimize the total violation. Columns with reduced cost equal to 0 are basic to ODYMP; only these columns are included in the Phase 1 problem. The reduced cost for column  $k$  is:  $RC^k = -Cost^k - (\pi_0 - \Lambda Fcap^k)$ , where  $\Lambda$  is the vector of bid prices and  $Cost^k$  is the total cost for  $k$ th FAM solution,  $Fcap^k$  and  $\pi_0$  is the dual of the convexity row. The weights on non-basic columns are 0, therefore, the Phase 1 problem only includes the basic columns. We use dual values,  $\rho$ , from the Phase 1 problem to

adjust bid prices to move in a direction that further reduces the violation. We can move in this direction until another column becomes basic.

The next column becomes basic when its reduced cost is 0. Let  $\Lambda'$  and  $\pi_0'$  be the bid prices and convexity dual after the next step:  $\Lambda' = \Lambda + \rho\delta$ ,  $\pi_0' = \pi_0 + \rho_0\delta$  where  $\delta$  is the step length,  $\rho_0$  is the dual on the convexity row in the Phase 1 problem. Let  $RC^{k'}$  be the reduced cost for column k after this step, the maximum step size will result in  $RC^{k'} = 0$ .

$$RC^{k'} = -Cost^k - \pi_0' + \Lambda' Fcap^k = RC^k - \rho_0\delta + \rho\delta Fcap^k$$

So,  $RC^{k'} = 0 \rightarrow \delta_k = \frac{RC^k}{\rho_0 - \rho Fcap^k}$ . The maximum step size is  $\delta_{\max} = \min_k(\delta_k)$ . The

additional basic columns are added to the Phase 1 problem. This process continues until an acceptably low level of violation is achieved on the capacity constraints. The algorithm is as follows.

1. Get initial bid prices,  $\Lambda$ . Use the final bid prices from the previous major iteration.

$$2. \text{ If } \frac{\partial rev_p(alloc_p)}{\partial alloc_p} > alloc_p \sum_{i \in p} \lambda_i \text{ then } alloc_p = \Gamma^{-1} \left\{ 1 - \frac{\sum_{i \in p} \lambda_i}{rev_p} \right\}, \text{ otherwise}$$

$$alloc_p = 0.$$

3. Get the dual value for the convexity row:  $\pi_0 = \max_k \{\Lambda Fcap^k - Cost^k\}$
4. Find  $K_0$ , the subset of columns such that  $\Lambda Fcap^k - Cost^k = \pi_0$ .
5. Solve the following Phase 1 problem to minimize the total allocation violations:

$$\min TotalViolation = \sum_l (s_l + t_l)$$

*ST*

$$\sum_{p \in l} alloc_p - \sum_k w_k cap_l^k + s_l - t_l = 0 \forall l \in L, k \in K_0$$

$$\sum_k w_k = 1, k \in K_0$$

$$w_k, s_l, t_l \geq 0$$

6. If  $TotalViolation \leq \varepsilon$  STOP

7. Determine maximum step size:  $\delta_{\max} = \min_{k \notin K_0} \left\{ \frac{-Cost^k - \pi_0 + \Lambda Fcap^k}{\rho_0 - \rho Fcap^k} \right\}$ .

$\rho_0$  is the dual on the convexity constraint for the problem solved in Step

5;  $\delta = \min(\delta_{\max}, \delta_{step})$ , where  $\delta_{step}$  is a default step length.

8. Update bid prices:  $\Lambda \leftarrow \Lambda + \rho \delta$ . Return to Step 2.

This algorithm achieves primal feasibility by driving capacity violations to 0. It achieves dual feasibility for the ODYM columns by setting allocations in Step 2. It achieves dual feasibility in the capacity columns by ensuring that the reduced cost for every column is 0 or negative.

### 7.3 Airline Planning Integration

Airline fleet assignment modeling is one step in a complex planning process. There are multiple phases to the planning process based on the time horizon, decision variables and constraints. Enterprise Planning addresses processes far in advance of operations.

Airlines make decisions associated with business objectives and the infrastructure required to achieve them. They are constrained by financial resources and regulation.

Product Planning determines how the airline is positioned in the marketplace in terms of schedule and price. The decisions made in Enterprise Planning provide the framework and the constraints for Product and Operational Planning. Finally, Tactics and Operations determine how many of each product is offered in the marketplace based on the schedule and pricing framework developed in the Product Planning stage. This process is summarized in Jacobs et al. (2000), see Figure 7.4.

To be most effective each of these planning stages should have a common goal, such as maximizing the net present value of future profits. In addition, each stage should anticipate the impact of its decisions on the subsequent stages. This approach creates a framework for hierarchical modeling. Incorporating revenue management into FAM is one example of this type of modeling. Hierarchical modeling has not been more widely used in airline planning due to modeling and/or computational limitations. It may become more feasible in the near future. While this process has been difficult to achieve to-date, improved communications within the airline, between airlines and their suppliers and between airlines and their customers should facilitate the development of more integrated decision making (Smith et al. 2001, Klabjan 2003).

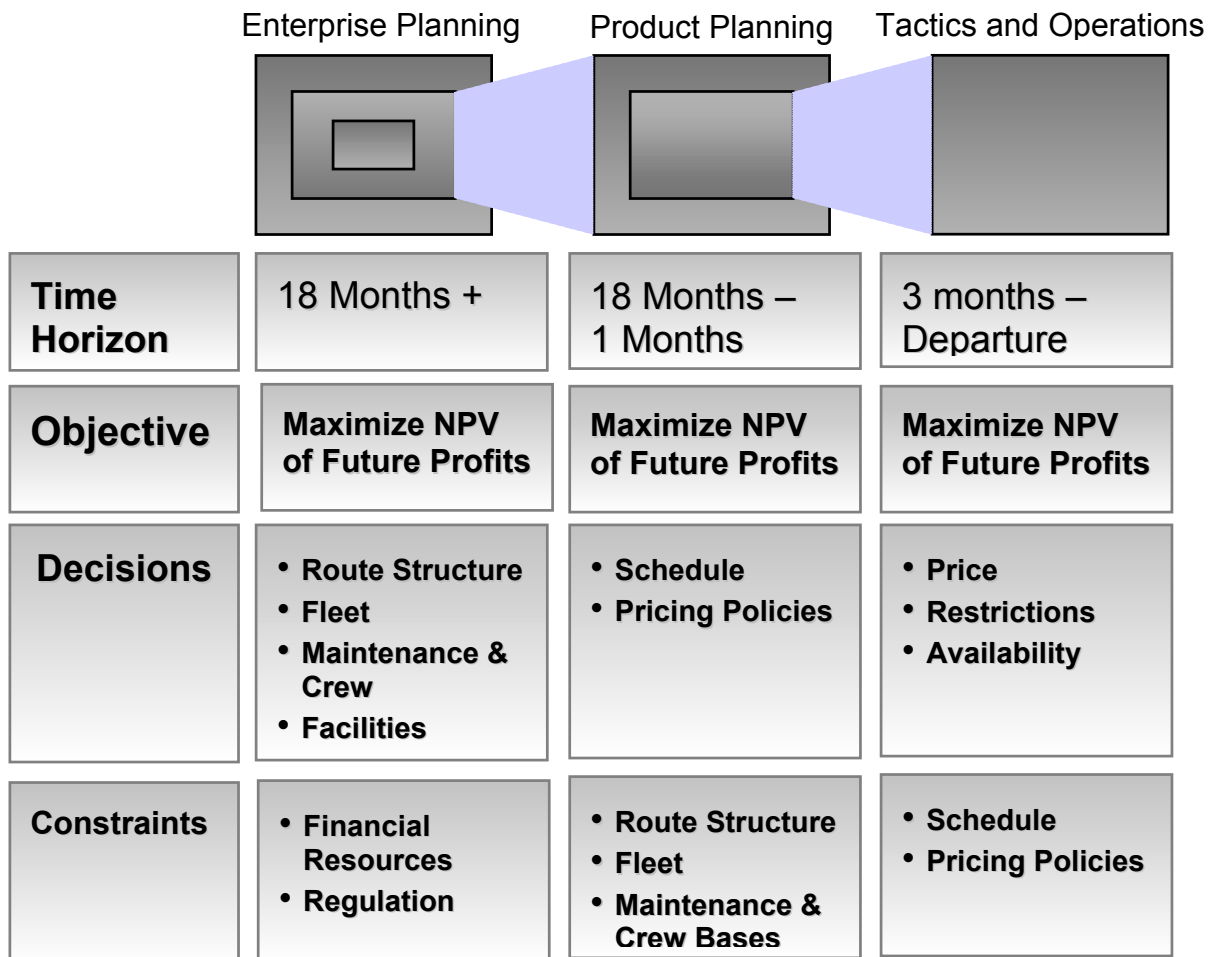


Figure 7.4: A Summary of airline planning stages

Potential areas of improvement in Enterprise Planning include:

- Labor planning. The feasibility and cost of fleet and/or route structure changes are affected by the impact on an airline's staff. The cost of hiring, training and operating depends on the current and future demographics of the airline's pilots, flight attendants, mechanics and ground staff. A labor planning model could be directly incorporated into Enterprise Planning models and processes as a sub-problem.
- Schedule planning. Airlines often analyze the future by building schedules for specific scenarios. Since this is a largely manual process few options can be evaluated. A scheduling component could be developed that is a sub-problem to the Enterprise Planning process.

Improvements in FAM related to Tactics and Operations include:

- Competitive planning. Airlines often assume that competitive schedules are fixed. As a result they are often overly optimistic about their future competitive position. An approach should be developed to allow airlines to analyze a more diverse and realistic set of future scenarios.
- Future RM approaches. The revenue management landscape is changing. Business practices are evolving as established carriers compete more directly with low-cost carriers. Revenue management forecasting is improving with the incorporation of customer choice modeling. Revenue management optimization using approximate dynamic programming is becoming more computationally feasible. These approaches should be incorporated in FAM and into Enterprise Planning.



## References

- Abara, J. (1989). "Applying Integer Linear Programming to the Fleet Assignment Problem." *Interfaces*, Vol. 19, No. 4, pp. 20-28.
- Ageeva, Y. (2000). "Approaches to Incorporating Robustness into Airline Scheduling." Masters Thesis, Operations Research Center, Massachusetts Institute of Technology.
- Ahuja, R. K., J. Goodstein, A. Mukherjee, J. B. Orlin, D. Sharma (2001). "Very Large-Scale Neighborhood Search Algorithm for the Combined Through and Fleet Assignment Model." MIT Sloan School of Management Working Paper 4388-01.
- Ahuja, R. K., J. B. Orlin (2002). "Very Large-Scale Neighborhood Search in Airline Fleet Scheduling." *SIAM News*, Vol. 35, Number 9.
- Barnhart, C., E. Johnson, G. Nemhauser, M. Savelsburg, P. Vance (1998). "Branch-and-Price: Column Generation for Solving Huge Integer Programs." *Operations Research*, Vol. 46, No. 3, pp 316-329.
- Barnhart, C., N. L. Boland, L. W. Clarke, E. L. Johnson, G. L. Nemhauser, R. G. Shenoi (1998). "Flight String Models for Aircraft Fleeting and Routing." *Transportation Science* 32 3 208-220.
- Barnhart, C., T. Kniker, and M. Lohatepanont (2002). "Itinerary-Based Fleet Assignment." *Transportation Science* 36 2 199-217.
- Barnhart, C. and A. Cohen (2002). "Airline Schedule Planning: Accomplishments and Opportunities." Working paper, Operations Research Center, Massachusetts Institute of Technology.
- Bazaraa, M. S., H. D. Sherali, C. M. Shetty (1979). *Nonlinear Programming Theory and Algorithms*. John Wiley and Sons.
- Beckman, J. M. (1958). "Decision and Team Problems in Airline Reservations." *Econometrica*, Vol. 26 pp 134-145.
- Belobaba, P. P. (1989). "Application of a Probabilistic Decision Model to Airline Seat Inventory Control." *Operations Research* 37, 183-197.
- Berge, M. E. and C. A. Hopperstad (1993). "Demand Driven Dispatch: A Method for Dynamic Aircraft Capacity Assignment, Models and Algorithms." *Operations Research*, Vol. 41, No. 1, pp. 153-168.
- Bian, F., E. K. Burke, S. Jain, G. Kendall, G. M. Koole, J. D. Landa Silva, J. Mulder, M. C. E. Paelinck, C. Reeves, I. Rusdi, M. O. Suleman (2003). "Making Airline

- Schedules More Robust.” Available from  
<http://citeseer.nj.nec.com/bian03making.html>.
- Boyd, E. A., I. C. Bilegan (2003). “Revenue Management and E-Commerce.”  
*Management Science*, Vol. 49, No. 10, pp 1363-1386.
- Clarke, L. W., C. A. Hane, E. L. Jonhson, G. L. Nemhauser (1996). “Maintenance and  
 Crew Considerations in Fleet Assignment.” *Transportation Science*, Vol, 30 No. 3 pp  
 249-260.
- Cots, R. B. (1999). “Revenue Management under Demand Driven Dispatch.” Masters  
 Thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of  
 Technology.
- Drake, A. W. (1967). *Fundamentals of Applied Probability Theory*. McGraw-Hill Inc.
- Du Merle, O., D. Villeneuve, J. Desrosiers and P. Hansen, P (1999). “Stabilized Column  
 Generation.” *Discrete Math*, Vol. 194, pp 229-237.
- DuMerle, O., J. P. Vial (2003). “Speeding Up Column Generation Via Convex  
 Nondifferentiable Optimization Tools: An Application to the Aircraft Rotation  
 Problem.” *AGIFORS Symposium Proc. 43*, Paris France.
- Erdmann, A., M. Kiahaschemi, A. Noltemeier and R. Schrader, R. (1997). “Fleet  
 Assignment with Respect to Itineraries.” Presentation at Mathematical Programming  
 Conference, Lausanne, Switzerland.
- Erdmann, A., A. Nolte, A. Noltemeier, R. Schrader (1999). “Modeling and Solving the  
 Airline Schedule Generation Problem.” Available from  
<http://citeseer.nj.nec.com/erdmann99modeling.html>.
- Etschmaier, M. M., D. F. X. Mathaisel (1984). “Aircraft Scheduling: State of the Art,”  
*AGIFORS Symposium Proc. 24*.
- Farkas, A. (1996). *The Influence of Network Effects and Yield Management on Airline  
 Assignment Decisions.*” PhD Dissertation, Department of Civil and Environmental  
 Engineering, Massachusetts Institute of Technology, Cambridge, MA.
- Feldman, J. M. (2002). “Matching Planes to People.” *Air Transportation World*,  
 December 2002.
- Ferguson, A. R. and G. B. Dantzig (1955). “The Aircraft Routing Problem,”  
*Aeronautical Engineering Review*, Vol. 14, No. 4.

- Ferguson, A. R. and G. B. Dantzig (1956). "The Allocation of Aircraft to Routes – An Example of Linear Programming Under Uncertain Demand." *Management Science* Vol. 3 pg 45-73.
- Gotz, S., S. Grothklags, G. Kliewer, S. Tschoke (2001). "A Local Search Based Approach for Large Scale Fleet Assignment Problems." Working paper, Department of Mathematics and Computer Science, University of Paderborn.
- Guenther, D. (1998). *Airline Yield Management*. PhD dissertation, Georgia Institute of Technology.
- Hane, C. A., C. Barnhart, E. L. Johnson, R. E. Marsten, G. L. Nemhauser, G. Sigismondi (1995). "The Fleet Assignment Problem: Solving a Large-scale Integer Program." *Mathematical Programming* 70 pp. 211-232.
- Hiriart, J. B., C. Lemarechal (1993). *Convex Analysis and Minimization Algorithms II Advanced Theory and Bundle Methods*, Springer-Verlag, Berlin, Germany.
- ILOG (2000). *ILOG Concert Technology 1.0 User's Manual*.
- ILOG (2001). *ILOG CPLEX 7.1 User's Manual*.
- ILOG (2001). *ILOG Solver 5.1 User's Manual*.
- Jacobs, T. L. (2004). "AA ODFAM Case Study." Technical report, American Airlines, unpublished.
- Jacobs, T. L., E. L. Johnson, B. C. Smith. (1999). "O&D FAM: Incorporating Passenger Flows into the Fleeting Process." *AGIFORS Symposium Proc. 39*, New Orleans LA.
- Jacobs, T. L., R. M. Ratliff, B. C. Smith (2000). "Soaring with Synchronized Systems." *ORMS Today*.
- Johnson, E. L., E. R. Barnes (1998). Class Notes from ISYE 6679 Computation Optimization. Georgian Institute of Technology.
- Hu, J., E. L. Johnson (1999). "Computational Results with a Primal-Dual Subproblem Simplex Method." *Operations Research Letters* Vol. 25 pp 149-157.
- Kang, L. (2003). *Degradable Airline Scheduling*, PhD Dissertation, Operations Research Center, Massachusetts Institute of Technology.
- Klabjan, D. (2003). "Large-scale Models in the Airline Industry." Working paper, Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign, Urbana IL.

- Kliwer, G. (2000). "Cooperative Approaches for Market Modeling and Fleet Assignment." Presentation to ISMP 2000, University of Paderborn, Germany.
- Knicker, T. S. (1998). *Itinerary-Based Fleet Assignment*. PhD Dissertation, Massachusetts Institute of Technology.
- Krishna, A., C. Barnhart, E. L. Johnson, D. Mahidara, G. L. Nemhauser, R. Rebello, A. Singh, and P. H. Vance (1995). "Advanced Techniques in Crew Scheduling." presented at INFORMS, Los Angeles CA.
- Lohatepanont, M. (2001). *Airline Fleet Assignment and Schedule Design: Integrated Models and Algorithms*. PhD Dissertation, Massachusetts Institute of Technology.
- Lee, A. O. (1986). *Airline Reservation Forecasting: Probabilistic and Statistical Models of the Booking Process*. PhD Dissertation, Massachusetts Institute of Technology.
- Lemarechal, C. (1989). "Nondifferentiable Optimization." G. L. Nemhauser, A. H. G. Rinnooy Kan, M. J. Todd, eds. *Handbooks in Operations Research and Management Science, Vol. 1, Optimization*. North-Holland, Amsterdam, pp 529-572.
- Lettovsky, L., E. Johnson, B. C. Smith (1999). "Schedule Generation Model." *AGIFORS Symposium Proc. 39*, New Orleans, LA.
- Littlewood, K. (1972). "Forecasting and Control of Passenger Bookings." *AGIFORS Symposium Proc. 12*, Nathanya, Israel.
- List, G. F., B. Wood, L. K. Nozick, M. A. Turnquist, D. A., Jones, E. A., Kjeldgaard and C. R. Lawton (2003). "Robust Optimization for Fleet Planning Under Uncertainty." *Transportation Research, Part E 39*, pp 209-227.
- Listes, O., R. Dekker (2002). "A Scenario Aggregation Based Approach for Determining a Robust Airline Fleet Composition." *Econometric Institute Report EI 2002-17*, Erasmus University Rotterdam, The Netherlands.
- McGill, J., G. J. van Ryzin (1999). "Revenue Management: Research Overview and Prospects." *Transportation Science*, Vol. 33, pp 233-256.
- Mead, K. M. (2003). "Airline Industry Metrics." U.S. Department of Transportation Memorandum, CC-2003-048.
- Pastor, J., (1999). "Demand Driven Dispatch: A Stochastic Approach." *AGIFORS Symposium Proc. 39*, New Orleans LA.
- Rexing, B., C. Barnhart, T. Kniker, A. Jarrah, N. Krishnamurthy (2000). "Airline Fleet Assignment with Time Windows." *Transportation Science*, Vol. 34, No. 1, pp 1-20.

- Rockafellar, R. T., R. J. Wets (1991). "Scenarios and Policy Aggregation in Optimization Under Uncertainty." *Mathematics of Operations Research*, Vol. 16, No. 1, pp 119-147.
- Rosenberger, J. M. (2001). *Topics in Airline Operations*, PhD Dissertation, Industrial and Systems Engineering, Georgia Institute of Technology.
- Rushmeier, R. A. and S. A. Kontogiorgis (1997). "Advances in the Optimization of Airline Fleet Assignment." *Transportation Science*, Vol. 31, No. 2, pp 159-169.
- Senne, E. L. F. and L. A. N. Lorena (2000). "Lagrangean/Surrogate Heuristics for p-Median Problems." *Computing Tools for Modeling, Optimization and Simulation: Interfaces in Computer Science and Operations Research*, Kluwer Academic Publishers, pp. 115-130.
- Simpson, R. W. (1978). "Airline Operations Class Notes." Massachusetts Institute of Technology, Unpublished.
- Simpson, R. W. (1989). "Using Network Flow Techniques for Origin-Destination Seat Inventory Control." Flight Transportation Laboratory Memorandum, M89-1, Massachusetts Institute of Technology.
- Smith, B. C. (1992) "ODYM it's Profitable and Practical." *AGIFORS Symposium Proc.* 32, Chicago Il.
- Smith, B. C. (2004). "The Continuing Evolution of Revenue Management." *Ascend*, 2004 Issue No. 1.
- Smith, B. C., R. M. Darrow and J. F. Liemkuler (1992). "Yield Management at American Airlines." *Interfaces*, Vol 22, pg 8-31.
- Smith, B. C., R. M. Ratliff and T. L. Jacobs (1997). "Optimal Load Factors: The Importance of Coordinated Scheduling, Pricing and Yield Management Decisions." *AGIFORS Symposium Proc.* 37, Bali Indonesia.
- Smith, B. C., J. Barlow, B. Vinod (1998). "Airline Planning and Marketing Decision Support: A Review of Current Practices and Future Trends," in *The Handbook of Airline Economics*, D. Jenkins (ed.), The Aviation Weekly Group of the McGraw-Hill Companies, New York, NY, pg 117-130.
- Smith, B. C., C. W. Penn (1988). "Analysis of Alternative Origin-Destination Control Strategies." *AGIFORS Symposium Proc.* 28, New Seabury MA.
- Smith, B. C., D. P. Guenther, B. V. Rao, R. M. Ratliff (2001). "E-Commerce and Operations Research in Airline Planning, Marketing and Distribution." *Interfaces*, Vol 31 pp 37-55.

- Subramanian, C. J., J. Lautenbacher, S. J. Stidman (1999). "Yield Management with Overbooking, Cancellations and No Shows." *Transportation Science* Vol. 34 pp 147-167.
- Subramanian, R., R. P. Scheff, J. D. Quillinan, D. S. Wiper, D. S. and R. E. Marsten (1994). "Coldstart: Fleet Assignment at Delta Air Lines." *Interfaces*, Vol. 24, No. 1, pp 104-120.
- Swan, W. M. (1983). "Traffic Losses at High Load Factors." *AGIFORS Symposium Proc.* 23, Olive Branch Mississippi.
- Talluri, K. T. and G. J. van Ryzin (2004). *The Theory and Practice of Revenue Management*. Kluwer Academic Publishers, Norwell Massachusetts.
- Talluri, K. T. and G. J. van Ryzin (1996). "A New Provably Optimal Bid-Price Technology," *AGIFORS Reservation and Yield Management Study Group Proceedings*, Zurich, Switzerland.
- Thengvall, B. G., J. F. Bard, G. Yu (2003). "A Bundle Algorithm Approach for the Aircraft Schedule Recovery Problem During Hub Closures." *Transportation Science*, Vol. 37, No. 4, pp 392-407.
- Vanderbeck, F (2000). "On Dantzig-Wolfe Decomposition in Integer Programming and Ways to Perform Branching in a Branch and Price Algorithm." *Operations Research*, Vol. 48, No. 1, pp 111-128.
- Vinod, B. (1995). "Origin-and-Destination Yield Management." *The Handbook of Airline Economics*, D. Jenkins (ed.), The Aviation Weekly Group of the McGraw-Hill Companies, New York, NY, pg 459-468.
- Vinod, B. and R. Ratliff (1990). "A Discount Allocation Optimization Method Using Stochastic Linear Programming." *AGIFORS Reservations and Yield Management Study Group Proc.*, Rome, Italy.
- Waldman, G. L. (1993). "A Study of the Practicality and Profit Enhancement Potential of Demand Driven Dispatch in Airline Hub Operations." Flight Transportation Report R 93-3, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology.
- Williamson, E. L. (1992). *Airline Network Seat Inventory Control: Methodologies and Revenue Impacts*. PhD dissertation, Massachusetts Institute of Technology.